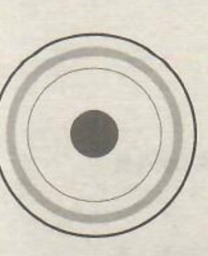


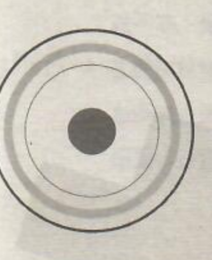


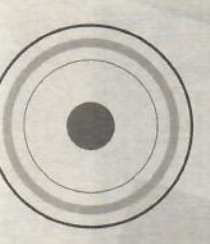


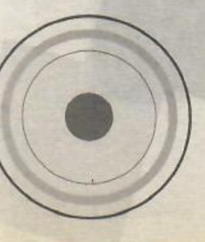


FreeBSD 入門應用

王俊斌　編著







FreeBSD 入門應用

作　　者：王俊斌
發 行 人：林麗芬
總 編 輯：許耀豪
出版統籌：劉慧楨
編　　輯：Jessica
出　　版：博碩文化股份有限公司
登 記 證：局版台業字第6261號
地　　址：台北縣汐止市新台五路一段112號10樓A棟
電　　話：(02) 2696-2869
傳　　真：(02) 2696-2867
網　　址：http://www.drmaster.com.tw
郵撥帳號：17484299
印 刷 廠：大然伊士曼彩色印刷股份有限公司
律師顧問：劉陽明
出版日期：西元2002年5月初版
ISBN號碼：957-527-484-9
博碩書號：OS20087

建議售價：460元

本書如有破損或裝訂錯誤，請寄回本公司更換

FreeBSD入門應用 / 王俊斌著. -- 初版 -- 台北縣汐止市：博碩文化，2002【民91】
面：　公分

ISBN 957-527-484-9(平裝)

1.作業系統

312.953　　91007918

筆者第一次接觸 FreeBSD 是退伍後剛進入大學時，由於對 UNIX 系統的興趣，偶然的發現在網路上有許多人討論 FreeBSD 這套作業系統，因此我才下定決心要學習 FreeBSD。剛接觸 FreeBSD 時，我連最基本的 UNIX 指令都不瞭解，在使用上也常常碰到問題，但總不因此而放棄。

在一開始時，每當我碰到問題時，第一步便是到 BBS 上的 BSD 群組發問，在台灣有許多 FreeBSD 的使用者，它們都非常熱心且樂意為新手解決問題。但每次遇到問題時就發問，雖然大都可以獲得解答，但一直問這種基礎的問題讓我覺得很浪費別人的時間。因此，筆者開始從系統中的 man page 及網路上現有的文章尋求協助，卻意外地發現資料異常地豐富。

在 FreeBSD 的社群中，有許多熱心的高手，他們總不吝於和人分享使用經驗。筆者對於高手們將自己的使用經驗放在網路上和他人分享十分佩服，因此，我也想效法他們的精神，將自己的學習過程製成網頁。原本筆者的網頁已幾近完成公開階段，但有感於目前市面上關於 FreeBSD 的書籍有限，且大多歷時已久，相對於其他 Linux 作業系統而言，FreeBSD 的能見度實在太低了。因此，我才決定將這些內容製成書籍。

對於初學者而言，學習新的作業系統難免會遭遇到一些瓶頸，但每當這些困難獲得解決時，就是一次新的成長。因此，當讀者使用上遇到困難時，千萬不要因而氣餒，而放棄了學好 FreeBSD 的機會。本書以初學者為導向，對於書中所提及的細節都經詳細的測試，以期使初學者能順利登上 FreeBSD 的殿堂。

讀者可以由本書架構發覺，本書將帶領讀者從基本的系統使用、網路伺服器架設到深入系統管理。筆者將個人在管理學校伺服器的經驗分享給讀者，希望對有心學習 FreeBSD 的使用者能有莫大的收獲。本書考量許多第一次接觸 FreeBSD 的使用者大多也是第一次接觸 UNIX 作業系統。因此，在本書的最後二個章節，說明了 UNIX 指令的應用，期望初學者能對 UNIX 系統的使用上有更多的認識。

本書的完成除了筆者本身外，還要感謝在筆者學習過程中曾經幫助過我的高手們，特別感謝中央研究院計算機中心張毓麟先生對於本書第十一章 Sendmail 設定的指導，更感謝 FreeBSD 提供我們這樣優秀的作業系統。

王俊斌 2002年3月30日 於中央大學



Chapter1 FreeBSD 簡介

Chapter2 安裝 FreeBSD

Chapter3 編譯核心

Chapter4 建立友善的介面

Chapter5 使用者管理

Chapter6 網路設定

Chapter7 /etc目錄下的檔案介紹

Chapter8 軟體安裝

Chapter9 X Window 的使用

Chapter10 網頁伺服器

Chapter 11 郵件伺服器

Chapter 12 DNS 伺服器

Chapter 13 NAT 及防火牆

Chapter 14 Proxy Server

Chapter 15 資料庫系統

Chapter 16 Samba 網路芳鄰

Chapter 17 系統安全

Chapter 18 指令應用

Chapter 19 Shell Script

附錄A 版權宣告

附錄B Ports 軟體分類列表

附錄C 製作 FreeBSD 安裝光碟

chapter 1 FreeBSD簡介

1.1 什麼是FreeBSD？

我想大家都知道 Microsoft Windows 是一套作業系統，FreeBSD 也是一套作業系統。FreeBSD 是一個可以在 Intel 相容個人電腦、DEC Alpha 或 PC98 架構的電腦上執行的 UNIX 作業系統。大家應該聽過另一套 UNIX 的作業系統 Linux，FreeBSD 也是一套免費的作業系統。它可以讓我們的個人電腦變成先進的工作站，更穩定地提供你所需的網路服務。

FreeBSD 作業系統相當容易取得及安裝，除了經由傳統的光碟安裝外，它也可以經由網路安裝、MS-DOS 分割區安裝等等。當然，我們也可以在電腦中同時安裝多種不同的作業系統，例如 Windwos 98 和 FreeBSD 同時並存也是件十分容易的事。

在 FreeBSD 上的應用軟體相當的多，也都可以免費取得，由於 FreeBSD 的穩定性高且功能強大，因此許多大型網站都以它為作業平台，其中最知名的就是 YAHOO!。Yahoo 是一個流量相當大的入口網站，他們選擇以 FreeBSD 為作業平台，由此可知 FreeBSD 的優異性。除此之外，在台灣，FreeBSD 普遍被應用於學術網路上，許多大專院校的伺服器都是使用 FreeBSD 來提供網路服務。

在寬頻網路逐漸普及的台灣，每個人都可以自行架設一台網路伺服器，以 FreeBSD 來提供網路服務(如網頁、郵件、檔案存取等)。值得一提的是 FreeBSD 並不像 MS-Windows 一樣每每要求使用者升級電腦才能使用。FreeBSD 對於硬體的要求很低，你可以用一台 Intel 586-133MHz 的舊電腦來安裝 FreeBSD ，這也算是廢物利用吧。



1.2 為什麼要選擇FreeBSD？

現在的個人電腦作業系統市場中，是以MS-Windows 獨大，但在網路伺服器市場中，UNIX 系統的使用率可不輸 MS-Windows 喔。我個人認爲 MS-Windows 之所以會有那麼多的使用者，主要是因爲它的使用者介面對初學者而言較容易操作，再加上許多軟體的配合及盜版的助長，安裝軟體只要一直按下一步就完成。使得使用者即便它的穩定性不高也得乖乖的接受。

其實初學者沒有試過其他的作業系統才會有這樣的誤解，因爲一直用 MS-Windows 才會認爲當機是無可避免的事，當機對於 MS-Windows 或許是無可避免，但在其他 UNIX 系統中可不常見。在 UNIX 系列的作業系統中，也可以有像 MS-Windows 的圖形介面，幾乎所有在 MS-Windows 上可以做的事，在 FreeBSD 上都可以做得到，唯一的不同點是你不必花錢去取得你想要的功能。包括排版、圖形處理、MP3、多媒體、網路芳鄰等等都可以在 FreeBSD 中做到。

說了這麼多，我想再和低效能的 windows 系統比較已經沒什麼義意了。然而，有這麼多的免費 UNIX 作業系統中，爲什麼要選擇 FreeBSD 而不使用其他作業系統(如 Linux)呢？在網路上在討論這個問題時，每每會引發每個作業系統使用者的激辯。對於要使用何種作業系統，除了使用者偏好外，還有許多指標可以提供我們參考。

BSD UNIX 系統可以說是網路作業系統的始祖，FreeBSD 是眾多 BSD UNIX 分支中的一個，它繼承了 BSD 系統的高性能與可靠性。自從 1993 年 FreeBSD 推出 1.0-RELEASE 以來，FreeBSD 開發團隊便致力於系統的調校，使其發揮絕佳的效能。在 FreeBSD 團隊的統籌努力下，使它比起

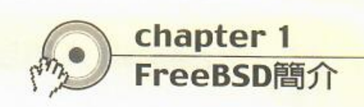
其他免費的 UNIX 作業系統更有結構。在 FreeBSD 上有許多支援的免費軟體，這些軟體大都已移植收錄於 FreeBSD ports 中，使得我們在安裝軟體時變得十分輕鬆。FreeBSD 是一套真正32 位元的作業系統，具有高效能核心架構、動態函式庫共享、絕佳的網路功能，比起其他商用 UNIX 系統毫不遜色。

我覺得 FreeBSD 總部統籌發展 FreeBSD 是一件很棒的事，所有的問題回報都可以統籌管理並予以更新。 FreeBSD 推陳出新的速度相當快，每一次安裝都確保這個版本不會有上一版的缺失。所以在 Linux 或 MS-Windows 中「新版本不一定是最好」的定律並不適用於 FreeBSD。但這並不意味著你必須不斷重新安裝系統，FreeBSD 總部隨時會發佈最新更新的檔案提供下載，而且如果使用 CVSUP 就可以和更新版的 FreeBSD 保持同步更新。

1.3 為什麼不選擇FreeBSD？

許多企業選擇使用目前大多數人使用的 MS-Windows 做爲一般作業用個人電腦的作業系統當然無可厚非。但以 MS-Windows 做伺服器，除了資訊人員的偏好外(或許因爲不會使用其他系統吧)，還有部份原因是爲了在企業內資訊人員技術不足時，能求助於系統供應商。而 FreeBSD 是免費的，企業也害怕有狀況時無人可以支援，這對企業是很重要的一項因素。

然而，FreeBSD 的使用人數其實很多，而且使用者都十分熱心，在台灣的 BBS 討論區上，許多問題都可以獲得解決。這當然還不夠，現在有很多顧問公司提供 FreeBSD 的顧問服務，可以提供企業這方面的服務。雖



然說 FreeBSD 的系統穩定，但在應用上如果沒有資訊人員的支援，很難能在企業中存活。我相信大部份的資訊人員都受夠了 MS-Windows 的折磨，這是我們該站起來的時候了，只有經由我們的主動學習，不屈不撓的精神，才能創造自己及企業的價值。

1.4 FreeBSD的版本命名規則

FreeBSD 每出一個新的版本都是以 FreeBSD A.B.C-TAG 來作爲命名的方式，例如 FreeBSD 4.5-RELEASE 或 FreeBSD 2.2.8-RELEASE。

- A-主要版本編號
- B-次要版本編號
- C-修正版本編號
- TAG-名稱標籤，如RELEASE、STABLE、CURRENT等

世界各地活躍的高手們組成 Core Team 對系統原始程式碼做開發和維護，幾乎系統原始程式碼每天都會有新版本和修正除錯。系統工具程式、驅動程式等等，例如 Ports 內的程式版本也常常更新。FreeBSD 的 Core Team 爲了兼顧發展新功能和穩定性，都會有一個實驗性的版本，以開發新功能爲主，稱爲 CURRENT ，而 FreeBSD 正式發行的版本稱爲 RELEASE 版本， 推出RELEASE之後會不斷的更新該版本以力求穩定性，稱之爲 STABLE。

目前最新的 RELEASE 版本是 FreeBSD 4.5-RELEASE，由一個 Team 負責，同時還有一個 Team 負責開發 FreeBSD 5.0-CURRENT。而 4.5-RELEASE 會不斷的更新，進而推出 STABLE 版本。 STABLE 及

CURRENT 推出時，在 FTP 中你會看到類似這樣的編號 FreeBSD 4.5-20020323-STABLE 或是 FreeBSD 5.0-20020321-CURRENT，其中 20020321 就是該版本推出的日期。

1.5 如何取得FreeBSD？

FreeBSD 可以經由免費取得，我們可以自各大 FTP 站台下載 ISO 檔，自行燒錄成光碟來安裝，也可以透過 FTP 站台以網路安裝。國內對 FreeBSD 收集最知名的應該是交大資工，如果您要下載 ISO 檔的話，可以在該 FTP 站台中的 ISO-IMAGE 目錄中找到。以交大資工而言，ISO-IMAGE 通常放在 /pub/i386/ISO-IMAGES/。我們一般的 PC 都是屬於 i386 的，所以您在 i386 目錄下看到一堆不同版本的 STABLE 及 RELEASE 都是你可以下載安裝的版本。

- 交大資工(ftp://freebsd.csie.nctu.edu.tw)
- 中央資工(ftp://freebsd.csie.ncu.edu.tw)
- 中研究(ftp://ftp2.tw.freebsd.org)
- 還有許多ftp1.tw.freebsd.org到ftp9.tw.freebsd.org都是在台灣的mirror站台

1.6 如何得到更多資訊？

FreeBSD 的使用者眾多，且十分熱心，在國內外有許多高手們將自己的經驗開放給使用者參考。以下即為部份 FreeBSD 的站台。

中文站台

- 台大電機Maxwell BBS BSD精華區 (telnet://bbs.ee.ntu.edu.tw)
- 中央大學企管系初心庭園BBS，386bsd版(telnet://bbs.ba.mgt.ncu.edu.tw)
- 台灣FreeBSD總站，交大資工FreeBSD (http://freebsd.csie.nctu.edu.tw)
- 中研院FreeBSD (http://freebsd.sinica.edu.tw)
- FreeBSD Chinese HOWTO(http://freebsd.sinica.edu.tw/zh-tut/t)
- 藍色泡泡的OHAHA (http://ohaha.ks.edu.tw)
- 大南國小的FreeBSD (http://freebsd.lab.mlc.edu.tw/)
- FreeBSD 使用大全 (http://tech.sina.com.cn/focus/FreeBSD/index.shtml)

英文站台

- FreeBSD總部 (http://www.freebsd.org)
- FreeBSD Handbook (http://www.freebsd.org/doc/)
- FreeBSD Cheat Sheets (http://www.mostgraveconcern.com/freebsd/)
- Fresh Ports (http://www.freshports.org/)
- ONLamp.com:FreeBSD Basics (http://www.onlamp.com/bsd/)

1.7 本書光碟使用說明

本書附有二片隨書光碟，第一片光碟爲 FreeBSD 4.5-RELEASE 安裝光碟，第二片爲本書所需的所有軟體及筆者設定檔的範例。

光碟二檔案說明：

表1

檔案名稱	說　明
/examples/	存放筆者的設定檔及一些範例程式。
/examples/etc設定檔/	筆者 /etc/ 目錄下的設定檔。
/examples/XWindow設定檔/	筆者關於 X Window 的設定檔。
/examples/adduser.tar.gz	大量新增帳號程式。
/examples/mysql.php	經由網頁管理 mysql 的工具。
/ports/distfiles/	存放本書所需軟體的原始檔。
/wintools/	MS Windows 下的工具。
/wintools/OS-BS.zip	多重開機管理員。
/wintools/putty.exe	在 MS Windows 下好用的 SSH 連線軟體。
/wintools/WinMD5.exe	檢查 MD5 的工具。

當您要以光碟安裝 FreeBSD 時，請使用光碟來開機。系統安裝完畢後，您可能會安裝一些可以在 FreeBSD 上使用的軟體。當我們使用 FreeBSD ports 安裝軟體時(參考第八章)，它會先檢查電腦中是否也有該軟體的原始檔，如果沒有則將自動從網路下載。爲了避免必須費時從網路下載，本書光碟二收錄了本書所提及的軟體，您可以在開始安裝其他軟體之前，所將光碟二 /ports/distfiles/ 目錄下的檔案複製到 /usr/ports/distfiles/ 目錄中。

如果您要將光碟二所有軟體的資料先存到硬碟中，請先將光碟二放入光碟機中，再執行下列指令來掛入光碟，並複製檔案。

```
# mount /cdrom
# cp -R /cdrom/distfiles/* /usr/ports/distfiles/
```

複製完畢後，如果要從光碟機中取出光碟，必須先執行下列指令才能將光碟退出：

```
# umount /cdrom
```

chapter 2 安裝FreeBSD

2.1 安裝前須知

2.1.1 如何取得FreeBSD

在這裡我們只介紹二種較常使用的安裝方式，即利用光碟安裝與經由網路安裝。如果要利用光碟安裝，可以使用本書所附之第一片光碟，其版本是 4.5-RELEASE。或者也可以到國內各大學的 FTP 站台取得 FreeBSD 的 ISO 檔來燒成光碟。國內對 FreeBSD 收集最知名的應該是交大資工，如果您要下載 ISO 檔的話，可以在該 FTP 站台中的 ISO-IMAGES 目錄中找到。以交大資工而言，ISO-IMAGE 通常放在 /pub/i386/ISO-IMAGES/。我們一般的 PC 都是屬於 i386 的，所以您在 i386 目錄下看到一堆不同版本的 STABLE 及 RELEASE 都是您可以下載安裝的版本，通常我選最新的 STABLE 版本來安裝，但通常做成 ISO 檔的大都是 RELEASE 版，所以如果要安裝最 STABLE 版，而且您的網路頻寬夠的話，可以用網路安裝。有時在 ISO-IMAGES目錄中有許多檔案，例如 4.5-install.iso 、4.5-mini.iso，那個 install.iso 就是我們要下載的檔，而 mini.iso 也是可以用來安裝的檔案，只是沒有一些常用的 packages ，所以檔案較小。以下為各主要 FTP 站台的網址：

- **交大資工** (ftp://freebsd.csie.nctu.edu.tw)
- **中央資工** (ftp://freebsd.csie.ncu.edu.tw)
- **中研究** (ftp://ftp2.tw.freebsd.org)

下載時您會發現該目錄中有一個檔案叫做 CHECKSUM.MD5，這是一個文字檔，用來檢查下載的檔案是否正確，您可以使用 FreeBSD 下的指令 md5 來檢查計算出來的結果和 CHECKSUM.MD5 中的記錄是否相同，

也可以使用 WinMD5 這個工具，在 MS Windows 計算。WindMD5 可以在本書所附光碟二的 wintools 目錄下找到。下載 ISO 檔後，把副檔名 .iso 改成 .nrg 後，再用 NERO 以燒錄映像檔方式，將該檔案燒錄成光碟，該光碟即具備光碟開機的能力。詳細的光碟燒錄方式，請參考本書附錄C的說明。

如果要以網路安裝的話，必須先做開機片，當然，您也可以使用類似版本的光碟來開機，再更改一些設定以安裝最新版本的 FreeBSD。要製作開機片，必須先到上述 FTP 站中下載三個檔案。

- pub/tools/fdimage.exe
- pub/i386/4.5-RELEASE/floppies/kern.flp
- pub/i386/4.5-RELEASE/floppies/mfsroot.flp

上面的各個目錄可能依 FTP 站台的不同有點變更，不過基本上 fdimage.exe 一定在 tools 的目錄下，而另外二個檔案一定在想要安裝的版本目錄下的 floppies 目錄中。上面目錄中的 4.5-RELEASE 是您想要安裝版本的目錄，您可以更改為想要的版本。下載完後，準備二張磁片，在 DOS 模式下執行下列指令以製作開機片。

```
c:>fdimage -f 1.44M kern.flp A:
c:>fdimage -f 1.44M mfsroot.flp A:
```

第一個指令完成後，第一張磁片就做好了，別忘了要先換第二張磁片後再下第二個指令，如此即完成開機片的製作。

2.1.2 安裝方式的取決

通常我會選最新版本的 STABLE 來安裝，但這種版本通常沒有 ISO 檔可以下載，在網路頻率不足時很難使用網路來安裝，除非自己事先下載最新的版本目錄下除了packages 目錄外的所有檔案來燒成光碟，否則就裝 RELEASE 吧，基本上差別不大，除非新的 STABLE 有修正重大的問題，不然的話差不多啦，您也可以在安裝完 RELEASE 後再升級。順道一提，如果您所下載的不是 ISO 檔，而是下載某一版本 FreeBSD 目錄下的檔案，想要自己燒錄可開機的光碟的話，您必須下載 floppies 目錄中的 boot.flp 作爲開機檔。

個人認爲同一版的 RELEASE 和 STABLE 不會有太大的差距，但有時碰到問題時，使用別的版本就可以解決。例如，我之前使用 4.2 的某一版本時，無法使用 ADSL ，換了很多版本都沒辦法，最後下載最新的 STABLE 來安裝就沒事了。並不會每次都這麼衰啦，也不會每次有問題換一個版本就沒事了。以前我都是以 STABLE 爲第一考量，但現在我覺得用 RELEASE 也不錯，因爲常看到最新的 STABLE 版本在 FTP 站中都不是很完整，可能有的目錄有問題或沒辦法安裝。如果網路頻寬夠的話，就用最新版本吧，無法安裝時再退而求其次。

2.1.3 硬碟分割表的概念

在 DOS 系統中，分割硬碟時，有一個主要磁區、一個延伸磁區，延伸磁區中再去分割成許多的邏輯磁區。在 FreeBSD 中，可將主要磁區分割成許多邏輯磁區 (logical slice)，做爲檔案目錄或虛擬記憶體。以我的電腦爲例，我有二顆IDE硬碟，在 FreeBSD 中的代號分別爲 ad0 及 ad1。在第

一個硬碟中，我有一個 DOS 系統及 FreeBSD 作業系統，這二個硬碟我分別再分割為許多邏輯磁區做為不同用途，因為的硬碟中有 DOS 系統已用去了一個主要磁區的話，那麼您看到第一個硬碟是 s2，舉例說明：

表2

ad0s1	ad0 代表第一個硬碟，s1 是第一個主要磁區，我放 MSDOS
ad0s2a	s2 代表第二個主要磁區，最後一個 a 在傳統上是指根目錄
ad0s2b	b 在傳統上是指 swap 虛擬記憶體
ad0s2e	/usr
ad1s1e	/home
ad1s1f	/var

最後一個 a 在傳統上是指根目錄；b 在傳統上是指 swap 虛擬記憶體；c 是指整個主要磁區；d 指的是整個硬碟；而 efgh 等是我們可以任意使用的，例如用來做 /usr 或 /home 。

2.1.4 硬碟空間的配置

要分割成多少個 slice 見仁見智，您可以把任何目錄都獨立成一個 slice (我想沒人這樣做)，如果 slice 分割成很多個的話，可能因為分割不良而造成日後有的空間滿了，有的沒用到；如果分割成很大，檔案都在同一個分割區上，當機時檔案遺失的機會比較大。

如果不同的目錄要設不同的參數的話，就要分割成不同的磁區，例如要架 BBS ，BBS 所使用的檔案都比較小，所以 inode會設比較多。所謂的 inode 指的是系統中可以建立的檔案及目錄總數，如果要儲存的檔案大部份都很小，則同樣大小的硬碟中會有較多的檔案，也就是說需要較多的 inode 來掛檔案及目錄。如果 inode 滿了而硬碟未滿的話還是不能儲存檔案，就好像是有一面牆(硬碟)，上面有很多勾子(inode)可以掛衣服(檔

案)。如果勾子少，則彼此的空間大，衣服大件一點的話很好，小件的話就會很空，浪費牆的空間。反之，如果勾子多則可以多掛點衣服，不過勾子也是會佔空間的，而且太多的 inode 會降低硬碟存取的效率。

如果不同目錄寫入的頻率不同，我會把較常寫入的目錄獨立出來，才不會影響其他檔案。又如果有多顆硬碟，我也會平均分配每個硬碟的使用頻率，利用分割的技巧來平均配置不同的目錄。

最簡單的分割方式就是一個根目錄和一個 swap，這樣就不會造成分割不良的浪費，不過這樣擋案毀損的機會也大。給大家我的分割方式供參考：

表3

目錄	大小	用途
/	200MB	根目錄，放開機必備檔，包含/bin、/etc、/tmp.....等
swap	128MB	虛擬記憶體，一般而言，都建議大小為記憶體的二倍，但現在記憶體實在太大了，我的 BBS 站有一百多人使用，記憶體有128MB，swap 分割成 128MB 就夠了。
/usr	2.6GB	放執行檔、設定檔等，日後安裝的軟體都會放在這裡，所以要大一點。如果要安裝X Window的話，2.5GB是差不多的，因為在安裝過程中，編譯時會用去很大的空間，安裝完清除後大概是1GB 吧。如果不安裝 X Window，只當伺服器用，那 1GB 一定足夠。
/var	500MB	這是放使用者信件、寄信時暫存區及一些系統記錄(log)的地方，如果信都不大、使用者不多，設成 100MB 也沒關係，如果要做 郵件伺服器的話，就設大一點吧。
/home	2.4GB	這裡是放使用者的目錄，我通常把網頁都放在這裡。如果有 BBS 的話，記得要獨立出來。

2.1.5 多重開機

FreeBSD 本身就支援多重開機，如果您想在電腦中安裝不同的作業系統，例如同時安裝 Windows 98 及 FreeBSD 的話，先安裝 Windows 98 再安裝 FreeBSD，在安裝 FreeBSD 時，可以選擇使用 FreeBSD 內定的多重開機程式，也可以安裝您喜歡的多重開機軟體。因爲 FreeBSD 的多重開機就只有二行字，畫面比較簡單，所以我會下載 OSBS 在安裝完後再在 Windows 98 下設定多重開機。這個也是見仁見智，如果不想使用 OSBS 的話，只要在等一下安裝時使用 FreeBSD 內定的開機管理員就可以了。

如果您使用了 FreeBSD或是 OSBS 的多重開機， 日後要移除多重開機管理員時只要在 MS-DOS 下執行下列指令即可：

```
c:\> fdisk /mbr
```

2.2 系統安裝

2.2.1 開機

如果使用光碟開機的話，只要放入本書的第一張光碟，就會進入設定核心的目錄選單，記得要先在 BIOS 中設定可以光碟開機。

如果使用磁片開機的話，放入安裝前須知中製作的磁片，一共有二張磁片，先放入 kern.flp 那一張，讀完之後會出現一個訊息，要您放入 mfsroot 那一張磁片，這時候就拿出 kern 那一張再放入 mfsroot 那一張磁片後，按 Enter 就可以繼續開機了。

2.2.2 設定核心

開機完後會出現下列的畫面：

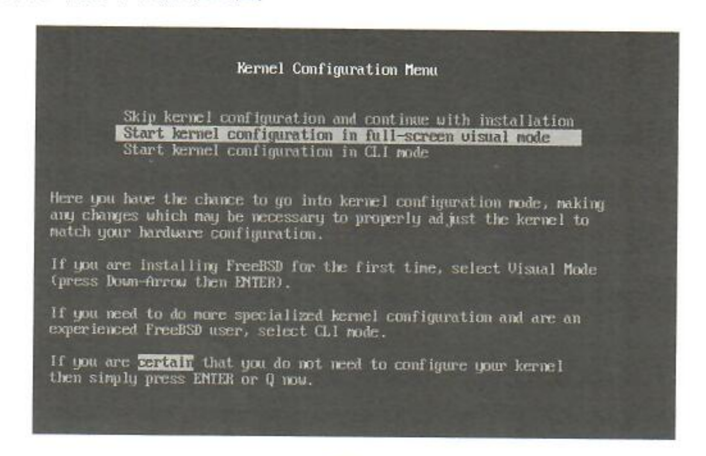


圖 2-1

這時候您可以選第一個跳過 kernel configure，也可以選第二個來設定核心。我通常第二個，因為有些硬體的驅動程式間會相衝，例如我的 SCSI 卡就不能選第一個直接進入，必須先移除其他的 SCSI 卡驅動程式。好吧，就選第二個「Start kernel configuration in full-screen visual mode.」

```
Active Drivers                          ? conflicts   Dev   IRQ   Port
Storage :                (Collapsed)
Network :                (Collapsed)
Communications :         (Collapsed)
Input :                  (Collapsed)
Multimedia :
Miscellaneous :          (Collapsed)

Inactive Drivers                                      Dev
Storage :
Network :
Communications :         (Collapsed)
Input :
Multimedia :
Miscellaneous :          (Collapsed)

[Enter] Expand device list     [X]  Expand all lists
[TAB]   Change fields          [Q]  Save and Exit          [?] Help
```

圖 2-2

在這裡分成上下二個視窗，最上面的 Active-Drivers 是您要使用的驅動程式，下面是您移除的。如果在上面移除的話，就會跑到下面來。您可以用 [TAB] 鍵在二個視窗中移動，使用 [DEL] 鍵來刪除設備。在這裡只要設定 Storage 及 Network 的選項即可。設定完後按 Q 存檔離開。

如果您的 SCSI 卡都是 PCI 介面的話，您可以把 Storage 中所有出現 SCSI 的選項都按[DEL] 來刪除，因為 PCI 的裝置都是即插即用，都自動抓得到。其他設備網路卡等也都是這樣，在 Network 選項中，裡面的設備都是給 ISA 介面用的，如果您的是 PCI 的網路卡的話，就把它們都刪了吧。如果您的設備有 ISA 介面的話，而您又不知道自己的設備是哪一個，就把它們都留著。

在 Storage 選項中，有 IDE 硬碟及軟碟機用的設備，請不要把它們刪除，最後的 Storage 選項至少應有下列圖2-3 所示的三個：

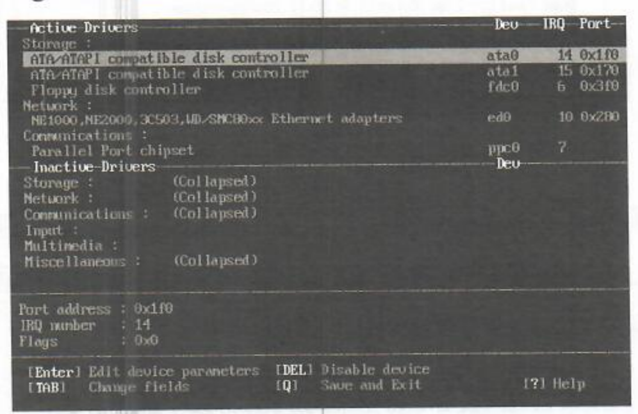


圖2-3

完成了上面的步驟，就可以按 Q 離開了，畫面會出現：

```
Save these parameters before exiting? ([Y]es/[N]o/[C]ancel)
```

按 Y 就可以離開，進入下一個步驟。這時候系統會一直跑，繼續開機的動作，最後會到一個藍色畫面。

2.2.3 開始自訂安裝

完成了開機後，會出現一個藍色畫面，這個畫面日後您可以在 /stand 目錄中，使用指令 sysinstall 去叫出這個安裝時的畫面。當開機進入藍色畫面時，您可以按 ALT+F2 來看除錯資訊，看一下是不是有硬體沒有安裝進來，由於 kernel 中只有安裝時要用到的硬體才會驅動，所以如果沒有抓到音效卡的話也沒關係。要回到藍色畫面就按 ALT+F1。

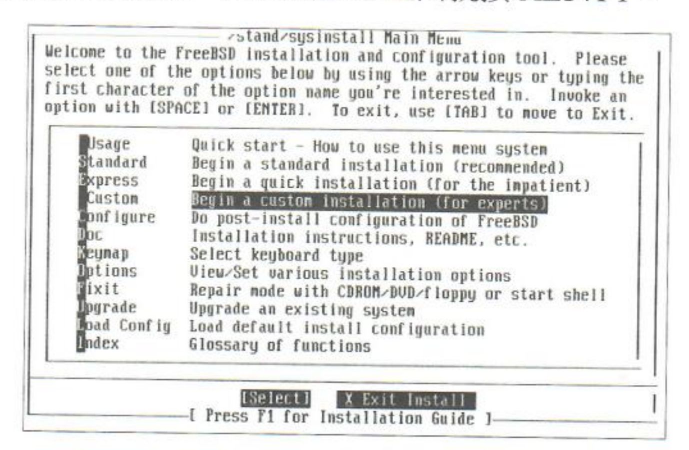


圖 2-4

我們選擇第四項 Custom 來自訂安裝。請使用上下鍵來選擇，並以空白鍵來確定進入。如果要把光棒從 Select 移到 Exit Install 的話，請使用左右鍵來移動。進入 Custom 後會出現下列畫面：

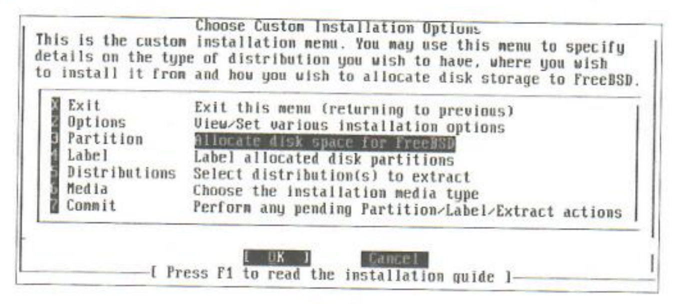


圖 2-5



如果您所使用的開機片並不是您想要安裝的版本的開機片的話，您要先選第二項 Options 進去修改 Release Name 的部份，否則就可以略過這一步，直接選 3 Partition 來選擇要安裝 FreeBSD 的硬碟磁區。

2.2.4 分割硬碟

進入 Partition 選單時，首先會出現硬碟選單，如圖 2-6 所示：

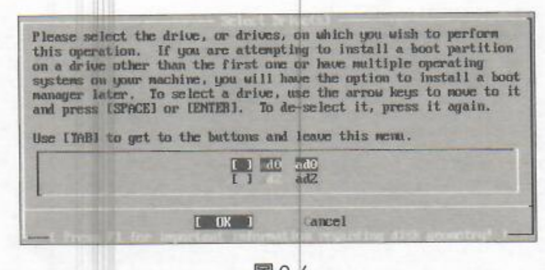


圖 2-6

您會看到上面的圖，我們之前說過 ad0 是第一個硬碟，而這裡的 ad2 是指第二個硬碟排線的第一個硬碟(master disk on the second IDE controller)。把光棒移到您要使用的硬碟，再按空白鍵進入即出現下面的畫面。如果二個硬碟都要使用，等一下分割完第一顆之後再來設定第二顆。

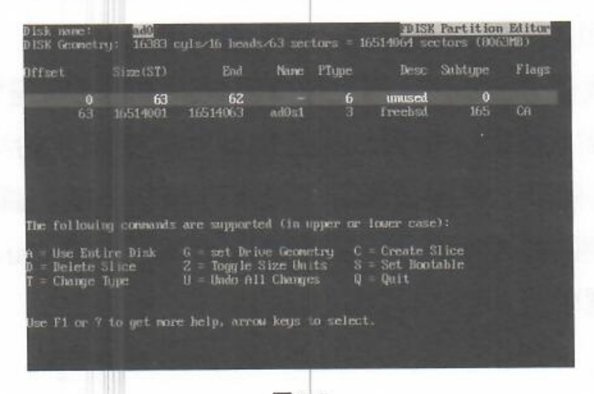


圖 2-7

在圖2-7中，光棒所指的就是未使用的空間，如果您這個硬碟只要給 FreeBSD 使用的話，您可以直接按 A 使用整個硬碟。如果您有 DOS 分割區的話，應該會出現一列是 fat 的磁區。把光棒移到 unused 的地方，按 C 來建立磁區，它會問您要使用多大的空間，您可以輸入 10000M 代表 10000 MB 也就是 10 GB 。接著按 Enter 鍵，它會問您 TYPE ，預設是165 就是按 Enter 就可以了。好了之後再對著剛設定好的 freebsd 磁區按 S 設定為可以開機。最後按 Q 離開畫面。離開後會出現下列畫面：

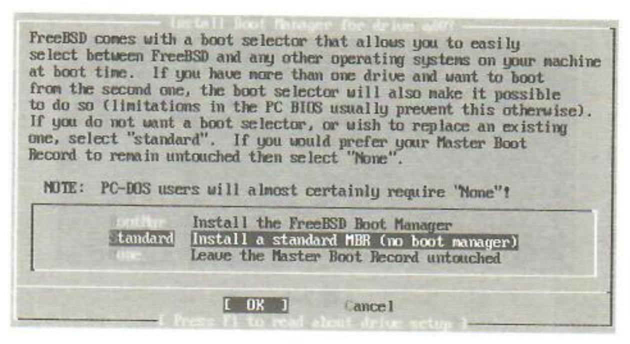


圖 2-8

上圖是問您要使用哪一種開機管理員，如果要使用其他作業系統，您可以使用 FreeBSD 的多重開機管理員，即選擇第一個 BootMgr 選項。如果要使用其他多重開機管理員就選 None，不過這樣一來如果沒有安裝其他多重開機管理員便無法使用 FreeBSD 開機。如果只有 FreeBSD 這個作業系統，就選中間這一個 Standard。我只要使用 FreeBSD ，並沒有其他作業系統，所以我選 Standard。選完後會回到剛才的硬碟選單，如果您還有其他的硬碟要加入的話，您可以選其他硬碟重複剛才的步驟，如果要加入的是 DOS 的硬碟，只要選擇進入後，什麼都不做，直接按 Q 離開即可。當所有的硬碟都設定好了，回到硬碟選單時，按 Cancel 離開選單，回到自訂安裝選單。



回到自訂安裝選單後，選第3個 Lable 進入 Disk Lable 編輯。即圖 2-9的畫面：

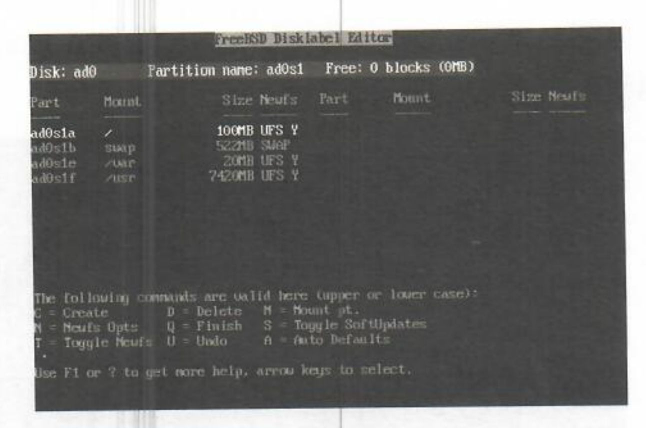


圖 2-9

我們之前在安裝前的須知中說過關於磁區的分割方式及大小配置，您可以直接按 A 自動配置，不過這樣出來的 swap 都會太大，如圖2-9中，swap 用了 522 MB。所以我們手動來做吧，最上方藍色光棒的位置就是您現在要分割的硬碟及其使用空間。如果有二個硬碟，所看到的就不是像上面那樣，而是有比較多個硬碟。不管硬碟有多少個，只要對著您要的硬碟按 C 去建立一個分割區，它會先問您大小，如圖2-10所示：

圖 2-10

在圖2-10詢問您所要分割的大小時，您可以輸入想要的空間大小，接著會出現圖 2-11 的畫面，詢問您要給哪一種類型的分割區，如果是要給目錄使用的話，就選 A file system，如果是 swap 就選 A swap partition。我們先來分割給根目錄使用，如下圖：

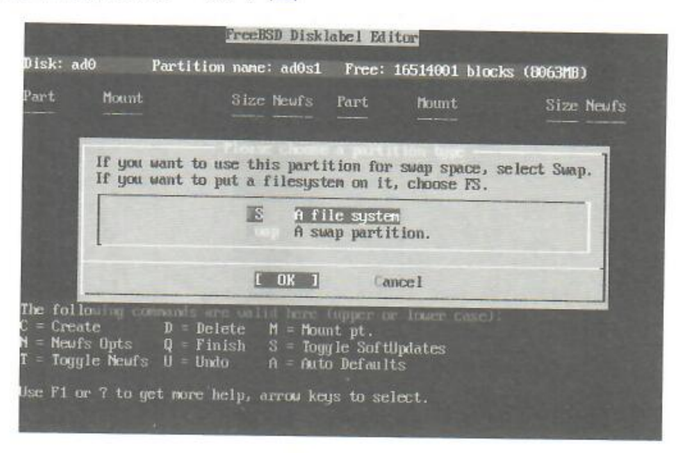


圖 2-11

接著選擇掛入點，如果是分割給 swap 使用的話，並不會問您掛入點是什麼。第一個分割區一定要先分割給根目錄，所以在這裡我們先分割根目錄，如圖2-12：

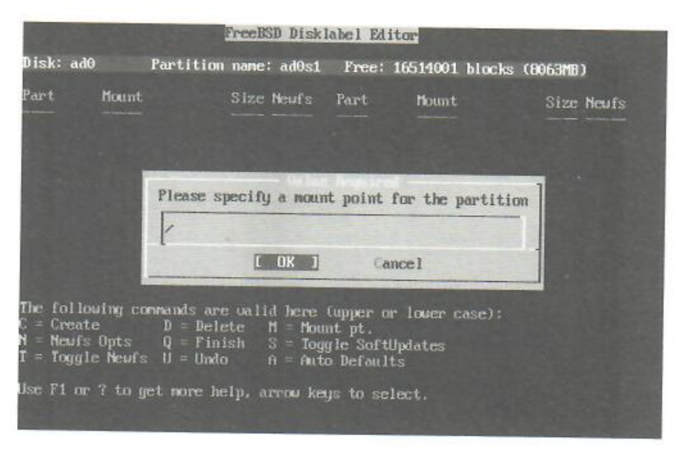


圖 2-12

這時您就輸入 / 就代表根目錄。接著再繼續分割給其他目錄，如 /usr、/home等。最後如圖2-13所示：

Part	Mount	Size	Newfs	Part	Mount	Size	Newfs
ad0s1	<none>	2000M	DOS				
ad0s2a	/	200M	UFS Y				
ad0s2b	swap	128M	SWAP				
ad0s2e	/usr	2600M	UFS Y				
ad0s2f	/var	500M	UFS Y				
ad0s2g	/home	2100M	UFS Y				
ad0s2g	/home/bbs	4000M	UFS Y				

圖2-13

在圖2-13中，如果您有 DOS 分割區在硬碟中，您可以把它掛進來(mount)做為一個目錄，這樣子在 FreeBSD 中就可以看到 DOS 分割區了。您可以對著最上面一行 DOS 那一個按 M ，並輸入目錄名稱為 /MSDOS ，就可以了。

如果您有 BBS 站的話，您必須更改一下 inode 的設定，才不會造成空間的浪費。把光棒移到 BBS 的目錄，再按 N 並輸入參數為 newfs -i 1024 -b 4096 -f 1024，來改變 inode 的設定。

好了，都設定好了就按 Q 離開，回到自訂安裝選單吧。接下來就開始選擇要安裝的套件。

2.2.5 安裝自訂套件

回到了自訂安裝選單後，選擇 Distribution 來選擇要安裝的套件。進入圖2-14的畫面後，選擇Custom：

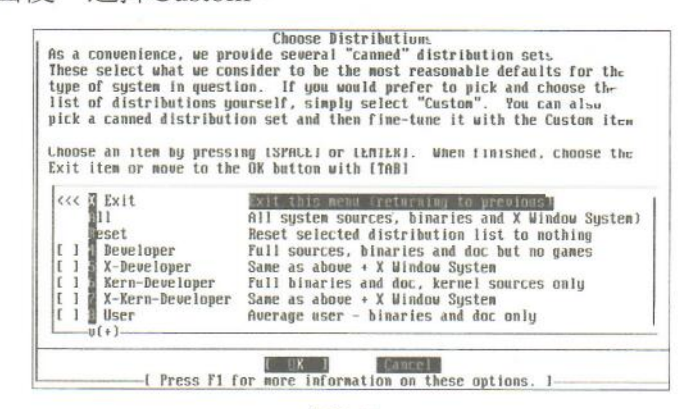


圖2-14

在上圖中選 Custom 後，會進入圖2-15的選單：

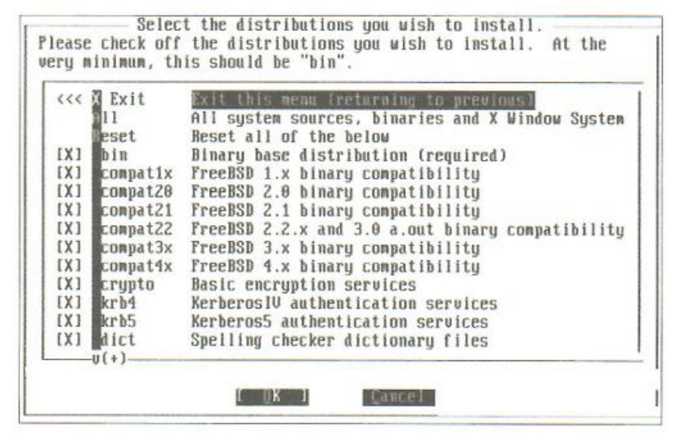


圖2-15

這時候除了最後二項 local 及 XFree86 外，全部都選，在選擇 src 選項時，會問您要選擇哪些東西，這時候就選 All 吧。現在的硬碟應該都不小，src 中的東西是一些原始碼，日後在修正一些 bug 時可能會用到。選

完之後就選 Exit 離開。回到上一層選單，再選 Exit 回到自訂安裝選單。接著就是選擇安裝的來源了。

2.2.6 選擇安裝來源

回到自訂安裝選單後，我們選擇 Media 來決定安裝的來源，如圖 2-16：

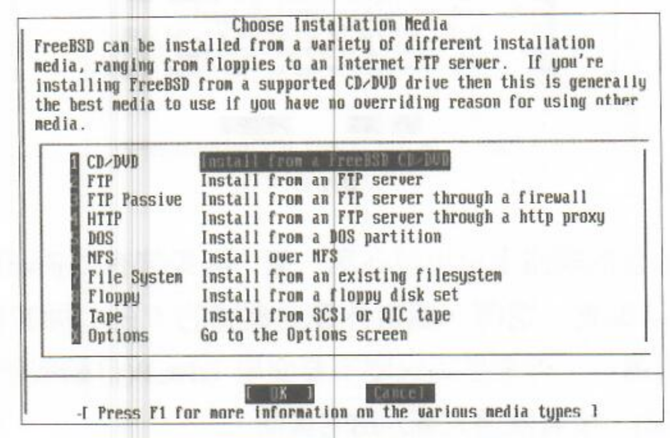


圖 2-16

如果您是以光碟安裝的話，就選第一個 CD/DVD 作爲安裝來源，選了之後就可以回到自訂安裝選單，選 Commit 開始安裝，並按下確定開始。

如果您是以網路安裝的話，就選第二個 FTP。選了 FTP 之後，會問您要使用哪一個 FTP 站台，這時候請選第二個 URL Specify some other ftp site by URL 來自訂要使用的 FTP 站台。進入後會問您要使用的站台位置，您要先去找出該 FTP 站台放 FreeBSD 的目錄爲何，以交大資工而言，您應該輸入：

```
ftp://freebsd.csie.nctu.edu.tw/pub/i386/
```

接著會要求您設定網路，使用網路安裝必須先設定網路，出現圖2-17的畫面，要您選擇網路卡：

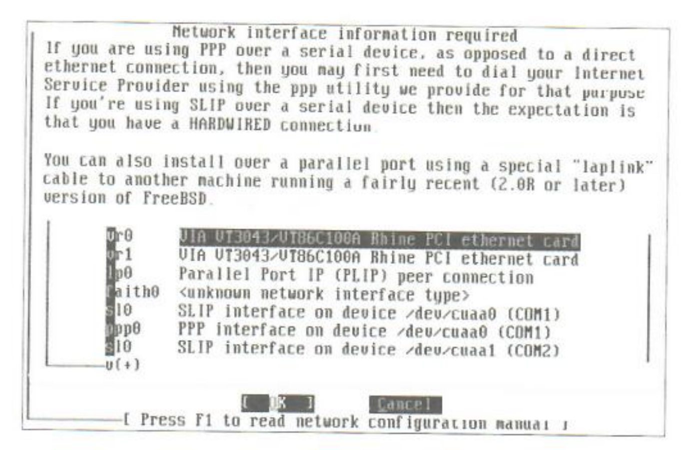


圖 2-17

圖2-17中，您的網路卡會出現在第一個，如果您有二張網路卡的話，就會有二個不同編號。選擇一個對外連到網路的卡後，接著會問您是否要使用 IPv6 請選否，再來是問您要不要使用 DHCP(自動取得 IP 位址....這就看您自己囉)，接著就進入圖2-18的畫面。

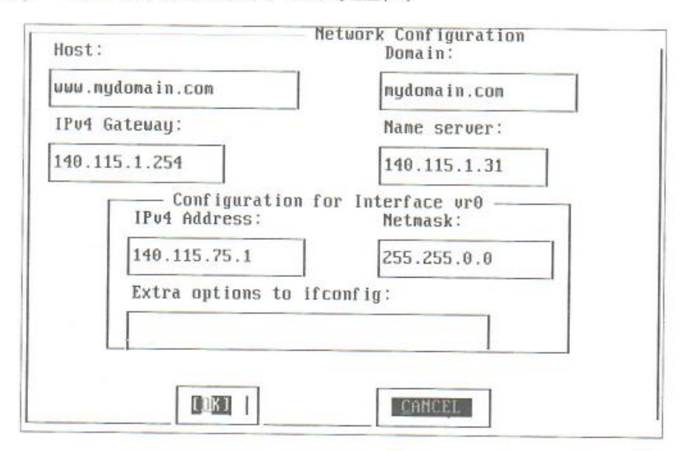


圖 2-18

您要先知道您的 Domain Name 及 IP 等，如果沒有 Domain Name 的話，就隨便輸入吧。假設我們的機器是 www.mydomain.com ，所以在Host一欄中就輸入 www.mydomain.tw，在Domain中就輸入 mydomain.com。我的機器在中央大學，所以 Gateway 就輸入 140.115.1.254，Name server 輸入 140.115.1.31，IPv4 Address 就輸入我的 IP 也就是 140.115.75.2，Netmask 也就是子網路就輸入 255.255.0.0。接著按 OK 離開回到自訂安裝選單。再來選 Commit 開始安裝，並按下確定開始。

2.2.7 最後的設定

經過了一段時間的安裝後，最後就會出現一個視窗問您是否要做最後的設定，這時候選要。又回到了一開始 sysinstall 的畫面。

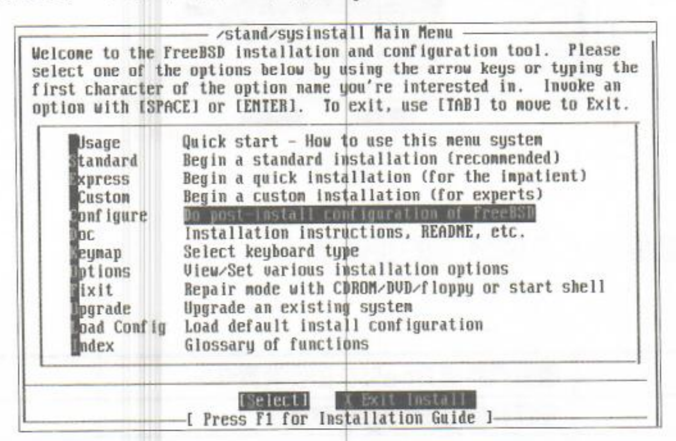


圖 2-19

這時候就選 Configure 進入設定選單，在 Configure 選單中，我們可以設定本機的基本資料。在這裡我們僅設定下列幾項：

- 設定網路
- 進入後選擇 Networking 設定網路，如果您是用網路安裝的話，您不必做，如果是用光碟安裝的話，就要。請參考前面安裝時選擇安裝媒體時所做的網路設定方式。
- 設定鍵盤速度及螢幕保護程式
- 設定時區 Time Zone

如果您要設定鍵盤速度，請選擇 Configure以進入設定選單，如圖2-20所示：

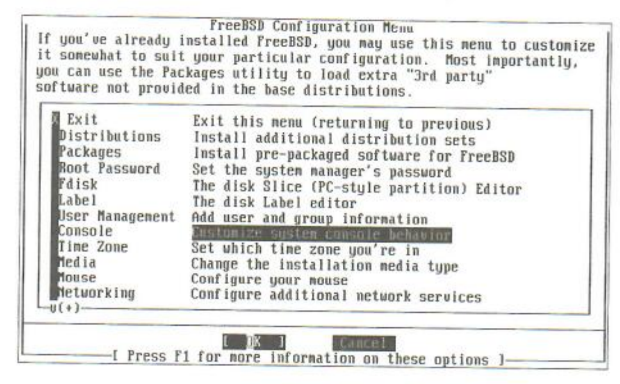


圖 2-20

這個設定選單中，各個選項的說明如下：

表4

Distributions	讓我們新增 FreeBSD 的安裝套件。
Packages	安裝其他 packages 軟體。
Root Password	設定超級使用者密碼。
Fdisk	分割硬碟的工具，日後要再加入新硬碟時可以使用。
Label	Disk Label editor，讓我們將已分割的硬碟再分割成不同磁區，以掛入 FreeBSD 中使用。
User Management	使用者帳號管理工具。
Console	文字模式下螢幕的設定。
Time Zone	設定時區。

Media	設定安裝軟體的來源。
Mouse	設定滑鼠。如果您要使用滑鼠，可以在此驅動。
Networking	設定網路。
Security	設定 FreeBSD Security Level。
Startup	設定開機時要啓動的選項，我們可以在 /etc/rc.conf 中設定。
TTYs	設定不同 TTY 的權限。
Options	設定一些安裝時選項。
XFree86	X Window 的設定。
Desktop	X Window 的桌面管理員設定。
HTML Docs	顯示 FreeBSD HTML文件。
Load LKM	從磁片中載入 kernel module。

這裡的設定只要先設定 Console 及 Time zone 即可，其餘的設定(如新增使用者及設定密碼等)我們都可以在重新開機後，於文字模式設定。

首先請選擇 Console 進入圖 2-21 的選單：

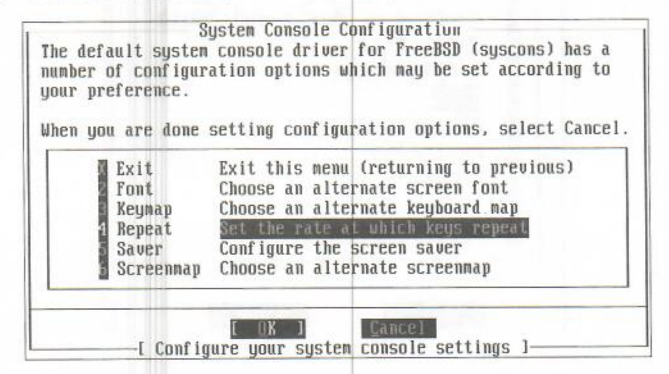


圖 2-21

在 System Console Configuration 選單中，第二及第三個選項 Font、Keymap 可以讓我們設定螢幕字型及鍵盤對映，不過我們不需要設定。第四項 Repeat 是設定鍵盤按鍵重覆的速度，因爲筆者喜歡按下一個鍵時，能快速的重覆輸入該鍵，所以我選了 Repeat 設定，並將速度設定 Fast。第五個選項是 Saver，我們可以選擇喜歡的螢幕保護程式，也可以選擇

Timeout 來設定啓動螢幕保護的時間。當您設定完成之後，請選擇 Exit 回到 Configuration Menu ，接著我們再選取 Time zone 來設定時區：

選擇了 Time zone 後，它會問您目前 BIOS 是否設定爲 UTC (格林威治時區)，如圖2-22所示，如果您不清楚，請選 NO。

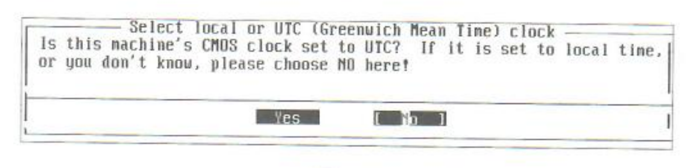


圖 2-22

接著要選擇所在區域，我們選擇第5項 Asia，如圖2-23：

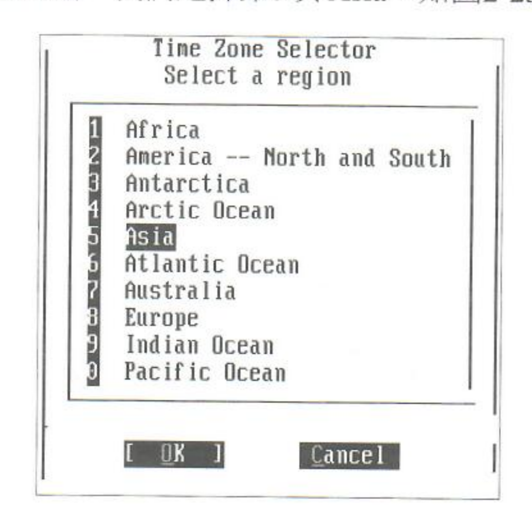


圖 2-23



然後再選擇國家 Taiwan 即可，如圖 2-24。

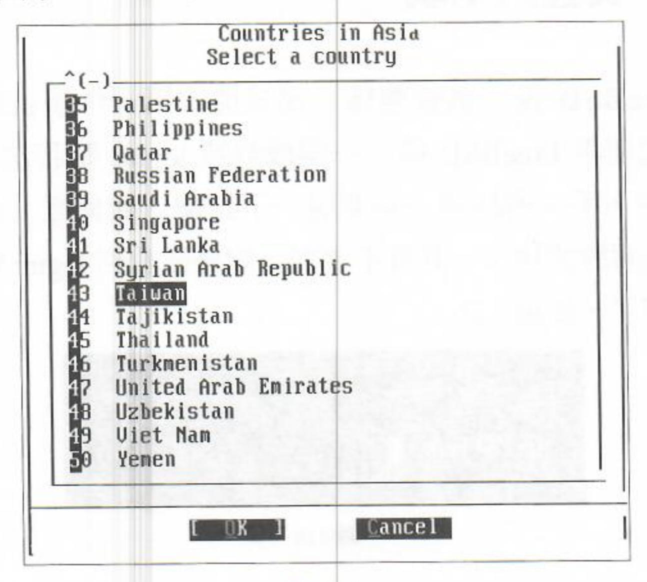


圖 2-24

最後系統會詢問您所選擇的 CST 時區是否正確，如圖 2-25，我們便選取 Yes 即可。

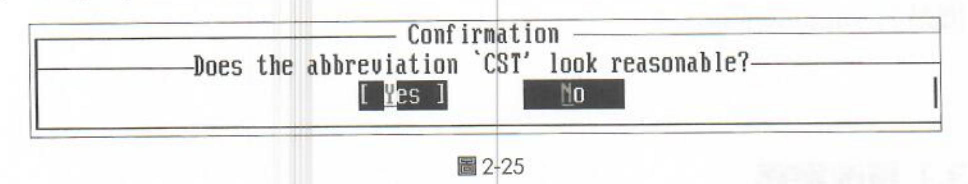


圖 2-25

我們在 Configuration Menu 所做的設定最後都會存放在 /etc/rc.conf 中，如果日後有需要更改，可以經由修改 /etc/rc.conf 來完成。

最後就一直選 Exit ，最後 Exit Install 離開安裝程式，重新開機即完成 FreeBSD 的安裝。如果您在過程中遇到問題，可以到各大 BBS 的 386bsd 版中發問。

2.3 第一次登入系統

安裝完 FreeBSD 後，重新開機，要知道的第一件事就是要怎麼使用嘛。我第一次使用 FreeBSD 時，一開機看到 login: 時我當場呆在那裡，完全不知如何下手。所以我一定要說一下這個，如何登入並更改密碼。說了一堆好像很難的樣子，其實不會啦，開機後看到 login: 時，打 root 就可以登入系統了，如圖2-26。

```
FreeBSD/i386 (mydomain.com) (ttyp0)

login: root
```

圖 2-26

由於是剛裝好的系統，所以不需要密碼。這裡的root 就是所謂的超級使用者，擁有所有權限。知道了如何登入，當然也要知道如何登出囉。您可以打指令 logout 就可以登出了。順便一提，要重開機就打 reboot 、要關機就打 shutdown now。

2.3.1 更改密碼

第一次登入後要更改 root 的密碼，請執行 passwd 指令來改密碼。系統會問您二次新密碼，以確認二次輸入的密碼相同。請不要使用太簡單的密碼，以免造成安全的漏洞。

2.3.2 新增第一位使用者

接著要設定第一個使用者的帳號，使用指令 adduser 來新增使用者。系統會問您一些問題：

```
輸入使用者的shell，我使用比較好用的 tcsh，故輸入 tcsh
Enter your default shell: csh date no sh tcsh [tcsh]:
使用者目錄要放在哪
Enter your default HOME partition: [/home]:
是否要從skel目錄中複製使用者的預設設定檔，直接按 Enter
Copy dotfiles from: /usr/share/skel no [/usr/share/skel]:
是否要送出訊息給使用者，直接按Enter，送出 /etc/adduser.message
Send message from file: /etc/adduser.message no  [/etc/adduser.message]:
是否要使用密碼，按 Enter ，預設是y
Use passwords (y/n) [y]:
使用者的帳號，只能使用小寫的英文字母及數字
Enter username [a-z0-9_-]:
使用者全名，輸入您的真實姓名
Enter full name []:
要使用的 shell，打 tcsh
Enter shell csh date no sh tcsh [tcsh]:
使用者的目錄，按 Enter 即可
Enter home directory (full path) [/home/john]:
使用者的編號，按 Enter 即可
Uid [1001]:
登入的等級，按 Enter 即可
Enter login class: default []:
登入的群組，按 Enter 即可
Login group john [john]:
```

```
是否要該使用者加入其他群組，請注意，因為第一個使用者是您自己，所以在這裡要
打 wheel ，這樣您才可以 su 成 root ，也就是您的帳號才可以登入成 root 帳號
Login group is `john". Invite asdf into other groups: guest no [no]:
密碼
Enter password []:
再輸入一次
Enter password again []:
以上都正確嗎？
OK? (y/n) [y]:
```

最後還有一些問題，都是直接按 Enter 就可以了，等到它問您是否要再增加下一個使用者時，回答 no 即可回到命令列。

設定到這裡後，就可以不必在機器前作業了。通常我的習慣是設定到這個地方後，就使用 telnet 的方式連線到機器來作其他的設定，這樣我就不必一定要呆在機器前，可以在自己家中使用別的機器連線到 FreeBSD 中做設定。

FreeBSD 自從 4.4-Release 起，預設就不開放使用 telnet 或 ftp 連線到機器中，如果您想使用 telnet 或 ftp 連線到機器，必須先編輯 /etc/inetd.conf，將 telnet 及 ftp 前的 "#" 拿掉，如下所示：

```
# $FreeBSD: src/etc/inetd.conf,v 1.44.2.6 2001/10/09 07:47:47 jkh Exp $
#
# Internet server configuration database
#
# Define *both* IPv4 and IPv6 entries for dual-stack support.
# To disable a service, comment it out by prefixing the line with '#'.
# To enable a service, remove the '#' at the beginning of the line.
#
```

```
ftp     stream tcp    nowait root  /usr/libexec/ftpd      ftpd -l
ftp     stream tcp6   nowait root  /usr/libexec/ftpd      ftpd -l
telnet  stream tcp    nowait root  /usr/libexec/telnetd   telnetd
telnet  stream tcp6   nowait root  /usr/libexec/telnetd   telnetd
#shell  stream tcp    nowait root  /usr/libexec/rshd      rshd
#shell  stream tcp6   nowait root  /usr/libexec/rshd      rshd
#login  stream tcp    nowait root  /usr/libexec/rlogind   rlogind
#login  stream tcp6   nowait root  /usr/libexec/rlogind   rlogind
```

接著，再使用下列指令來重新啓動 inetd 服務：

```
# kill -HUP `cat inetd.pid`
```

請注意這裡 `cat inetd.pid` 所使用的 ` 是鍵盤左上角的那一個符號，而非單引號。不過現在大家都建議使用 ssh 連線以取代 telnet，因爲 ssh 是以加密過的方式連線，比較不會以明碼的方式傳送資料。您可以使用 putty 這個軟體在 MS windows 下連線到 FreeBSD 中。您可以從 http://www.chiark.greenend.org.uk/~sgtatham/putty/ 下載 putty，也可以在本書光碟二的 wintools 目錄下找到該軟體。

您也可以在FreeBSD中使用 ssh 的方式連到別台機器

```
# ssh jack@123.456.78.9
```

這個指令表示以使用者 jack 身份連線到 123.456.78.9，我們也可以使用主機名稱的方式，例如jack@dns.abc.edu.tw ，或者也可以只打 ssh dns.abc.edu.tw 來登入，此時登入名稱會是您現在用的使用者名稱。

如果所連線的站台是第一次使用 SSH連線，則會出現下列一堆東西，表示接收到所連線站台RSA key，並詢問您是否要繼續連接。此時打 "yes" 三個字即可：

```
The authenticity of host '123.456.78.9' can't be established.
RSA key fingerprint is 13:96:8a:61:31:cf:32:3f:7a:0a:77:ad:7e:49:e7:3f.
Are you sure you want to continue connecting (yes/no)?  yes
```

系統會將所接收到的金鑰 (key) 存放在使用者家目錄下的 .ssh/known_hosts 檔案中。如果日後在 known_hosts 目錄中所記錄的站台金鑰有變更時，我們可以編輯該檔案以刪除舊的金鑰。

2.3.3 基本指令介紹

爲了一開始的使用，在這裡我們先說明一些簡單的指令用法，以利之後的設定，更詳情的指令介紹請參考本書第十八章「指令應用」，或使用指令 man 來查詢指令的用法。如果您是 UNIX 初學者，先閱讀「指令應用」可以讓您對 UNIX 指令及系統管理有更多的了解。

表5

cd	改變所在位置，和 DOS 中的用法類似，如果要到 /etc 下，就打 cd /etc。
pwd	直接打就會顯示目前所在目錄的名稱。
ls	看目錄中的檔案清單。打 ls 或 ls /etc，就像 DOS 中的 dir。
w	查詢目前在線上的使用者
ee	文書編輯軟體
date	看目前的日期及時間
mkdir	建立一個目錄，mkdir abc 就可以建立一個目錄叫 abc 。
rm	刪除檔案或目錄。刪除檔案時 rm file.txt，刪除目錄時要加參數 -rf ， 如 rm -rf abc
cp	複製檔案，就像 DOS 中的 copy。cp file newfile
man	查詢線上使用手冊，如 man cp 就可以看到 cp 這個指令的詳細說明。
su	變成超級使用者即 root。

2.3.4 FreeBSD 的目錄結構

在 Windows 作業系統中，在檔案總管中可以看到 Windows 的 "樹狀" 目錄結構。而 FreeBSD 中的目錄也是像一顆樹，一個目錄下還有很多個目錄，和 Windows 不同的是在 UNIX 系統中，每一個目錄都有一定用途。我們了解 FreeBSD 目錄結構的用意就是讓我們知道每個目錄的用途，日後我們要安裝新軟體或使用 FreeBSD 時，能按照這種規則來做，這樣一來在管理維護上比較方便，目錄也會比較有條理。

以下我們就簡單的說明 FreeBSD 的目錄結構，您也可以使用指令 man hier 來查看目錄結構說明。

表6

/	UINX 系統的根目錄，是目錄的最上層。
/bin/	放置基本的使用者指令，是開機時必備的。
/boot/	系統開機時必需用到的設定。
/dev/	UNIX 系統將週邊設備視為檔案來管理，這個目錄就是放置裝置節點檔 (device node)。
/dev/	MAKEDEV 就是用來管理這些節點檔的工具。
/etc/	放置系統的設定檔，例如使用者密碼、群組等。
/etc/defaults	放置預設的系統設定檔。請 man rc。
/etc/gnats/	gnats 的設定檔，請 man send-pr。
/etc/isdn/	isdn 的設定檔，請 man isdnd。
/etc/	kerberosIV/ kerberos version IV 的設定檔，請 man kerberos。
/etc/mail/	Sendmail 的設定檔。
/etc/mtree/	目錄權限的設定檔，請 man mtree。
/etc/namedb/	DNS 伺服器的設定檔，請 man named。
/etc/periodic/	每天、每週、每月定時要執行的設定，請 man periodic。
/etc/ppp/	ppp 的設定檔，請 man ppp。
/etc/ssl/	OpenSSL 的設定檔。
/etc/uucp/	uucp 的設定檔，請 man uucp。
/kernel	開機時系統會載入的核心 (kernel)。

/modules/	Kernel 可以載入的模組，請 man kldstat。
/mnt/	空目錄，我們可以用它來作為暫時 mount 檔案系統。
/proc/	系統執行中程序 (process) 資料，請 man procfs mount_procfs。
/root/	超級使用者 root 的家目錄。
/sbin/	系統程序及管理工具的目錄。
/stand/	這是安裝磁片上的指令。
/tmp/	暫存目錄，許多程式都會需要暫存目錄來存放資料。開機時會清除。
/usr/	包含主要的使用者工具及應用軟體。您可以把它看成 Windows 中的 windows目錄及 program file 目錄的集合。
/usr/bin/	一般的使用者指令及應用軟體。
/usr/games/	一些小游戲。
/usr/include/	標準 C 語言的標頭檔。
/usr/lib/	系統函式庫。
/usr/libdata/	一些系統工具的資料庫。
/usr/libexec/	系統服務程式 (daemons) 及工具。
/usr/local/	非 FreeBSD 所附的軟體都會安裝在這個目錄下，我們在安裝軟體時最好都安裝在這個目錄。您可以將它視為 Windows 作業系統中的program file 目錄。這個目錄中也有 bin sbin etc lib 等目錄。
/usr/obj/	在編譯 FreeBSD 系統時存放過程中暫存檔的位置。
/usr/ports/	FreeBSD ports 移植軟體的原始程式目錄，我們可以從這個目錄中找到自己想要的軟體來快速安裝。
/usr/sbin/	可以讓使用者執行的系統服務及工具。
/usr/share/	系統軟體共享的資料庫。
/usr/src/	放置 BSD 或其他軟體原始程式碼的目錄。
/usr/X11R6/	X Windows 的目錄。
/var/	放置系統記錄檔、暫存檔的目錄。
/var/account/	使用者執行過的指令記錄檔，請 man acct。
/var/at/	定時執行排程的資料檔。請 man at。
/var/backups/	系統重要檔案的備份區。
/var/cron/	使用者排程的資料表，請 man cron。
/var/db/	重要的系統資料庫。
/var/games/	內附的遊戲紀錄檔。
/var/log/	系統記錄檔，我們可以在這裡查看系統狀況記錄。
/var/mail/	使用者信件暫存區。
/var/preserve/	文件編輯時異常中止時，會將文件存到這個目錄，請 man ex。

/var/msgs/	系統訊系的資料庫，請 man msgs。
/var/quotas/	檔案系統使用容量限制的記錄。
/var/run/	記錄系統開機後執行狀態的暫存區。請 man utmp。
/var/spool/	列表機或郵件輸出時的緩衝區。
/var/tmp/	系統暫存區，開機時不會清除。
/var/yp/	the NIS maps。

chapter 3 編譯核心

3.1 為什麼要重新編譯核心

安裝完成時，所使用的核心是一般性的核心，稱之爲 GENERIC kernel。爲了要支援常見的軟硬體，因此 GENERIC 核心中可能包含了許多我們用不到的驅動程式，也可能不支援一些特殊的硬體。

在硬體方面，核心中包含了太多的東西不僅會佔去記憶體的空間，不同程式間也有可能造成衝突。GENERIC 中並未包含音效卡的驅動程式，因此如果你有音效卡的話，也必須重新將音效卡的驅動程式編譯進去。

在軟體方面，如果要啓動 FreeBSD 的防火牆功能，或是使用 ADSL 連線(4.4 以前的版本)，都需要重新將支援這些功能的參數加到核心中。另外，如果要改變系統的效率，你必須修改核心中的參數，例如增加同時上線的人數、或最大同時開啓的檔案數等。當然，有的功能在 FreeBSD 中可以經由 sysctl 這個指令來修改，而毋需修改核心，但大部份的功能是一定要修改核心的。所以我們還是要來了解一下如何爲自己量身訂做一個新的核心。

請放心，編譯核心並不難，其實只有幾個步驟，只要依下列的方法去做，相信您對於系統核心將有更深入的了解。



3.2 修改核心

首先，您必須確認你在安裝 FreeBSD 時，有將 src 裝進來。cd /usr/src/sys/i386/conf/ 確定該目錄存在，這個目錄中有二個檔案，一個是 GENERIC，一個是 LINT。GENERIC 就是安裝時用的一般核心，而 LINT 是完整的核心及說明。如果你對核心中的某一個參數有任何問題，可以去 LINT 中找它的說明。以音效卡爲例，由於 FreeBSD 主要的用途是作爲網路伺服器，因此內定的核心 GENERIC 中並未包含音效卡的設定。因此如果我們要使FreeBSD支援音效卡的話，我們必須從 LINT 中找出正確的音效卡設定來加入核心中。

首先，我們要做的就是將 GENERIC 複製一份，並修改複製的那一份。

```
# cp  GENERIC  MYKERNEL
```

上面那一行指令就是把 GENERIC 複製一份叫做 MYKERNEL，新的 KERNEL 要叫什麼名字你可以自己取。接著就要來修改 MYKERNEL 了，用文書處理軟體打開 MYKERNEL 後，再依文件中的說明來修改，把不必要的移除。必須要注意的是，核心中有的參數是互相依賴的，也就是說某些參數的存在一定要有某一行的支援。現在使用指令 ee 來修改剛才複製的核心。

```
# ee MYKERNEL
```

在核心中，如果在一行的開頭有"#"表示該行爲註解，以下我們針對 GENERIC 核心中每一個參數的說明。

3.2.1 基本的設定

```
# 這一行是必備的，代表的是使用 i386 的機器，也就是 IBM 相容個人電腦
machine     i386

# 這裡是指 CPU 的類型，最近的 CPU 都是 I686_CPU
# 有些 CPU 如 Cyrix 233Hz CPU 或 Pentume Pro 也都是 686 的
# 我們可以使用 dmesg 指令來看 CPU 類型
cpu     I386_CPU
cpu     I486_CPU
cpu     I586_CPU
cpu     I686_CPU

# 這個可以說是核心的名字，把 GENERIC 改成 Alex 或 BBS 或 webserver
ident   GENERIC

# 這是用來控制系統內部表格(internal system tables)大小的參數，這個值一定要
# 設定大於四，maxusers 的值決定了處理程序所容許的最大值，20+16*maxusers
# 就是你將得到的所容許處理程序系統開機就必須要有 18 個處理程序 (process)，
# 即便是簡單的執行指令 man 又會產生 9 個 process，所以將這個值設為 64 應該是
# 一個合理的數目。如果你的系統會出現 proc table full 的訊息的
# 話，就把它設大一點，例如 128。
# 注意：maxusers 的值和同時可以上線的最大使用者人數並無關係，
# 它只是決定你的 kernel 中一些資料結構的大小。
# 真正影響上線人數的是 pseudo-device pty 16 。
maxusers     32
```

3.2.2一般選項

```
#makeoptions DEBUG=-g       #以 gdb 除錯模式來編譯核心
# 這個參數讓 kernel 用軟體的方式模擬浮點運算，如果你的 CPU 不
# 含浮點運算器 (或沒有 387)，你就必須打開此參數。
# CPU 等級是 486-DX 以上的人可以不需要這一行。
# 注意: FreeBSD 所提供的一般浮點模擬器並不是十分精準，如果你沒有浮點
# 運算器卻又需要最好的準確度，建議你把這一行改成GPL_MATH_EMULATE
# 來使用 GNU 浮點模擬。因為 GNU 版權的關係，因此不以它來當作內定的模擬器。
options        MATH_EMULATE

# 網際網路溝通協定，就算你不打算連上網路也要保留這一行
# 因為大多數的程式都會要求 lookback 的網路
options        INET
options        INET6   #IPv6 溝通協定

# 處理程序檔案系統，讓你可以使用像 ps 這個指令去得到執行中的
# 處理程序的資訊。
options        PROCFS      #Process filesystem
```

3.2.3 各種檔案系統的支援

```
# 最基本的檔案系統支援，如果你是從硬碟開機的，你一定需要它。
options         FFS                 #Berkeley Fast Filesystem
options         FFS_ROOT            #FFS usable as root device [請勿刪除]
options         SOFTUPDATES         #Enable FFS soft updates support

# 記憶體映對檔案系統 (Memory-mapped Filesystem)。基本上這是為了達到快速暫存
# 用的 RAM disk，當你有許多 swap 空間的時候這很有用。/tmp 是一個掛上 MFS
#  的好地方，因為許多程式會利用/tmp 建立暫時檔案。
options         MFS                 #Memory Filesystem
options         MD_ROOT             #MD is a potential root device

# 網路檔案系統(Network Filesystem)，除非你想要經由網路存取
# 其他工作站的檔案，否則你不需要它。
options         NFS                 #Network Filesystem
options         NFS_ROOT            #NFS usable as root device, NFS required

# MS-DOS 檔案系統。除非你每次開機都要使用 MSDOS 檔案系統不然你可以將它
# 拿掉，系統會在你使用到 MSDOS 檔案系統時自動載入。除此之外，你也可以使用
# mtools來存取 DOS 的軟碟這並不需要有 MSDOS 檔案系統的支援。
options         MSDOSFS     #MSDOS Filesystem
# ISO 9660 是 CD-ROM 的檔案系統，如果你只是偶爾用到 CD-ROM
# 你可以將它拿掉，系統會在使用到 CD-ROM 時自動載入。此外
# 用 CD-ROM 聽 Audio CD 不需要 CD9660 的支援。
options         CD9660              #ISO 9660 Filesystem
options         CD9660_ROOT         #CD-ROM usable as root, CD9660 required
```

3.2.4 軟硬體相容性設定

```
# 和 BSD 4.3 相容[請勿刪除!!]
options         COMPAT_43

# 在找 SCSI 設備時要延遲多久(MILLISECONDS)有的 SCSI 需要一段時間來起始，
#如果你有 IDE 的硬碟可以開機，那麼你可以乎略這一行。如果你是以 SCSI 硬碟
# 開機的你可以把這一行的數字調小一點以加快開機的速度。當然，如果因此
# 抓不到硬碟，你就必須再把這個值調高一點
options         SCSI_DELAY=15000

# 允許使用者抓取console，對 X window 使用者尤其有用。例如可以使用 xterm -C 來
# 模擬 console，以得到 talk,write 的訊息，或是任何從系統發出的訊息
options         UCONSOLE

# 讓你在開機時可以做些設定
options         USERCONFIG  #boot -c editor
# 這個選項讓你可以在開機時從開機選單啟動虛擬參數編輯
options         VISUAL_USERCONFIG#visual boot -c editor

# 允許核心除錯追蹤
options         KTRACE          #ktrace(1) support

# 這個參數提供 System V 共享記憶體的支援。最常使用 SYSVSHM 的
# 是 X Windows 的 XSHM 功能，有些圖形化的程式用使用它來提高執行
# 速度。如果你使用 X Windows 或是 BBS，你一定要打開此參數。
options         SYSVSHM
```

```
# 提供 System V semaphores 的支援，雖然不常用到，但是它只佔一
# 點點 kernel 的空間。如果你架 bbs 站，這是一定要的啦。
options         SYSVMSG
# 提供 System V messages 的支援，雖然不常用到，但是它只佔一點
# 點 kernel 的空間。如果你架 bbs 站，這是一定要的啦。
options         SYSVSEM
# Real-time extensions added in the 1993 POSIX. 有些程式(如 star office)會用到它。
options         P1003_1B                  #Posix P1003_1B real-time extensions
options         _KPOSIX_PRIORITY_SCHEDULING
# 這個選項限制 ICMP 封包錯誤回應，可減少被denial of service packet 攻擊的風險
options         ICMP_BANDLIM              #Rate limit bad replies

# install a CDEV entry in /dev
options         KBD_INSTALL_CDEV

# 如果要支援多個 CPU 的話，要把下列二行前的 # 拿掉
#options        SMP             # Symmetric MultiProcessor Kernel
#options        APIC_IO         # Symmetric (APIC) I/O
```

3.2.5 匯流排及軟碟機

```
device isa      #支援 ISA 匯流排
device eisa     #支援 EISA 匯流排，在 586 以後就沒用到了
device pci      #支援 PCI 匯流排

# 軟碟機，如果只有一個軟碟機，可以把 fd1 拿掉
device fdc0     at isa? port IO_FD1 irq 6 drq 2
device fd0      at fdc0 drive 0
device fd1      at fdc0 drive 1
#
# 如果你使用的是 Toshiba Y-E Data PCMCIA 軟碟機，
# 不要使用上面那一個 fdc0 那一行，改用下面這一行：
#device         fdc0
```

3.2.6 IDE 介面裝置

```
# ATA and ATAPI devices，就是 IDE 介面的東西
device ata0     at isa? port IO_WD1 irq 14
device ata1     at isa? port IO_WD2 irq 15
device ata
device atadisk # IDE 硬碟 (ATA disk drives)
device atapicd # IDE 光碟機 (ATAPI CDROM drives)
device atapifd  # IDE 軟碟,如 MO( ATAPI floppy drives)
device atapist  # IDE 磁帶機 (ATAPI tape drives)
options         ATA_STATIC_ID           # 靜態的分配裝置代號
```

3.2.7 SCSI 裝置

```
# SCSI Controllers，SCSI 介面的控制
# 以下是不同廠牌的 SCSI 卡，可以使用 dmesg 指令表看你的 SCSI 卡是哪一種型號
device ahb      # EISA AHA1742 family
device ahc      # AHA2940 and onboard AIC7xxx devices
device amd      # AMD 53C974 (Tekram DC-390(T))
device isp      # Qlogic family
device ncr      # NCR/Symbios Logic
device sym      # NCR/Symbios Logic (newer chipsets)
options         SYM_SETUP_LP_PROBE_MAP=0x40
# Allow ncr to attach legacy NCR devices when
# both sym and ncr are configured
device adv0     at isa?
device adw
device bt0      at isa?
device aha0     at isa?
device aic0     at isa?
device ncv      # NCR 53C500
device nsp      # Workbit Ninja SCSI-3
device stg      # TMC 18C30/18C50

# SCSI 週邊設備
device scbus    # SCSI bus (一定要有)
device da       # Direct Access 直接存取(硬碟)
device sa       # Sequential Access 循序存取(如磁帶等)
device cd       # 光碟機
```

```
device pass     # Passthrough device (direct SCSI access)
# RAID controllers interfaced to the SCSI subsystem
# SCSI 的磁碟陣列裝置
device asr      # DPT SmartRAID V, VI and Adaptec SCSI RAID
device dpt      # DPT Smartcache - See LINT for options!
device mly      # Mylex AcceleRAID/eXtremeRAID

# RAID controllers,SCSI 的磁碟陣列裝置
device aac      # Adaptec FSA RAID, Dell PERC2/PERC3
device ida      # Compaq Smart RAID
device amr      # AMI MegaRAID
device mlx      # Mylex DAC960 family
device twe      # 3ware Escalade
```

3.2.8 基本週邊設備

```
# atkbdc0 控制了鍵盤及 PS/2 滑鼠
device atkbdc0 at isa? port IO_KBD
device atkbd0   at atkbdc? irq 1 flags 0x1
device psm0     at atkbdc? irq 12

# VGA 顯示卡
device vga0     at isa?

# 啓動時更新螢幕，螢幕保護程式會使用到它
pseudo-device splash
```

```
# syscons is the default console driver, resembling an SCO console
# 螢幕
device sc0      at isa? flags 0x100

# 支援 vt200 相容終端機
# Enable this and PCVT_FREEBSD for pcvt vt220 compatible console driver
#device         vt0     at isa?
#options        XSERVER     # support for X server on a vt console
#options        FAT_CURSOR  # start with block cursor
# If you have a ThinkPAD, uncomment this along with the rest of the PCVT lines
# 如果你有 IBM ThinkPAD 筆記型電腦，或其他使用 IBM 鍵盤的，就要使用這一行
#options        PCVT_SCANSET=2      # IBM keyboards are non-std

# 浮點數運算，一定要有
# Floating point support - do not disable.
device npx0     at nexus? port IO_NPX irq 13

# 電源管理
# Power management support (see LINT for more options)
device apm0     at nexus? disable flags 0x20 # Advanced Power Management
# PCCARD (PCMCIA) support，PCMCIA 支援，筆記型電腦才有吧
device          card
device          pcic0   at isa? irq 0 port 0x3e0 iomem 0xd0000
device          pcic1   at isa? irq 0 port 0x3e2 iomem 0xd4000 disable

# Serial (COM) ports，序列埠，就是 COM1 COM2 COM3 COM4
```

```
device          sio0    at isa? port IO_COM1 flags 0x10 irq 4
device          sio1    at isa? port IO_COM2 irq 3
device          sio2    at isa? disable port IO_COM3 irq 5
device          sio3    at isa? disable port IO_COM4 irq 9

# Parallel port，並列埠，就是一般舊型列表機會用的連接埠
device          ppc0    at isa? irq 7
device          ppbus           # Parallel port bus (required)
device          lpt             # 印表機 Printer
device          plip            # 在並列埠上使用 TCP/IP
device ppi      # 並列埠介面的裝置
#device         vpo     # Requires scbus and da

# The probe order of these is presently determined by i386/isa/isa_compat.c.
device ie0      at isa? port 0x300 irq 10 iomem 0xd0000
#device         le0     at isa? port 0x300 irq 5 iomem 0xd0000
device lnc0     at isa? port 0x280 irq 10 drq 0
device cs0      at isa? port 0x300
device sn0      at isa? port 0x300 irq 10
```

3.2.9 網路卡設定

```
# PCI介面的網路卡 PCI Ethernet NICs.
# 你可以使用 dmesg 去看你的網路卡是哪一個型號
device de       # DEC/Intel DC21x4x (``Tulip")
device txp      # 3Com 3cR990 (``Typhoon")
device vx       # 3Com 3c590, 3c595 (``Vortex")

# 以下也是 PCI 介面的網路卡，使用MII bus，如果你的網路卡型號
# 是下列其中之一，請勿將 miibus 移除
# PCI Ethernet NICs that use the common MII bus controller code.
# NOTE: Be sure to keep the 'device miibus' line in order to use these NICs!
device miibus   # MII bus support
device dc       # DEC/Intel 21143 and various workalikes
device fxp      # Intel EtherExpress PRO/100B (82557, 82558)
device pcn      # AMD Am79C97x PCI 10/100 NICs
device rl       # RealTek 8129/8139
device sf       # Adaptec AIC-6915 (``Starfire")
device sis      # Silicon Integrated Systems SiS 900/SiS 7016
device ste      # Sundance ST201 (D-Link DFE-550TX)
device tl       # Texas Instruments ThunderLAN
device tx       # SMC EtherPower II (83c170 ``EPIC")
device vr       # VIA Rhine, Rhine II
device wb       # Winbond W89C840F
device wx       # Intel Gigabit Ethernet Card (``Wiseman")
device xl       # 3Com 3c90x (``Boomerang", ``Cyclone")
# ISA介面的網路卡，如果很不幸的你的網路卡是 ISA 介面，
```




```
# 建議你換成 PCI 的，否則你應該要先知道你使用的 irq 及 port
# 如果你的型號是 ed0 ，則請勿移除上述的 miibus
# 'device ed' requires 'device miibus'
device ed0      at isa? port 0x280 irq 10 iomem 0xd8000
device ex
device ep
device fe0      at isa? port 0x300
# Xircom Ethernet
device xe

# 無線網路,PRISM I IEEE 802.11b wireless NIC.
device awi
# 使用PCMCIA設備的無線網路
# WaveLAN/IEEE 802.11 wireless NICs. Note: the WaveLAN/IEEE really
# exists only as a PCMCIA device, so there is no ISA attachment needed
# and resources will always be dynamically assigned by the pccard code.
device wi
# Aironet 4500/4800 802.11 wireless NICs. Note: the declaration below will
# work for PCMCIA and PCI cards, as well as ISA cards set to ISA PnP
# mode (the factory default). If you set the switches on your ISA
# card for a manually chosen I/O address and IRQ, you must specify
# those parameters here.
device an
```

3.2.10 虛擬裝置

```
# loop 是 TCP/IP 的通用 loopback 裝置。這是必須的
pseudo-device loop      # Network loopback

# 如果你有乙太網路卡，這個也是必須的，用以支援乙太網路
pseudo-device ether     # Ethernet support

# sl 用以支援 SLIP，SLIP已經幾乎被 PPP 所取代
# ppp 可以更簡單、快速的建立 modem-to-modem 連線
# sl 後面的數字是要模擬多少條 SLIP
pseudo-device sl        1

# 支援撥接連線
pseudo-device ppp       1        # Kernel PPP

#tun 會被 PPP 所用
pseudo-device tun                # Packet tunnel.

# 最大的 ttys，如 telnet 同時上線最大人數，內定是 16
# 你可以在 pty 的後面加上數字來提高人數
# 最大是 256
pseudo-device pty                # Pseudo-ttys (telnet etc)
pseudo-device md                 # Memory "disks"
pseudo-device gif                # IPv6 and IPv4 tunneling
pseudo-device faith     1        # IPv6-to-IPv4 relaying (translation)

# 用來支援封包的監聽，你可以使用 tcpdump 來查看封包
pseudo-device bpf                #Berkeley packet filter
```

3.2.11 USB 裝置

```
# Pseudo devices - the number indicates how many units to allocate.
# 支援 USB
# USB support
device uhci          # UHCI PCI->USB interface
device ohci          # OHCI PCI->USB interface
device usb           # USB Bus (required)
device ugen          # Generic
device uhid          # "Human Interface Devices"
device ukbd          # Keyboard
device ulpt          # Printer
device umass         # Disks/Mass storage - Requires scbus and da
device ums           # Mouse
device uscanner      # Scanners
# USB Ethernet, requires mii
device aue           # ADMtek USB ethernet
device cue           # CATC USB ethernet
device kue           # Kawasaki LSI USB ethernet
```

3.2.12 音效裝置

在原本的 GENERIC 設定中並未將音效卡的驅動程式放入，如果您需要加入音效卡，您可以參考 LINT 中關於音效卡的選項。如果您的音效卡是 PIC 或支援即插即用(PnP)，在設定上會比較簡單，以 SoundBlaster 16、64、128 等為例，只要加入下列選項：

```
device      pcm
device      sbc
```

如果沒有支援 PnP，如 ESS1868 則使用下列設定：

```
device      pcm
device      sbc0    at isa? port 0x220 irq 5 drq 1 flags 0x15
```

如果您的音效卡還是無法驅動，建議您參考 LINT 中的設定，找出適合您的音效卡。在安裝完音效卡後，您必須在 /dev/ 中增加音效卡的裝置節點：

```
# cd /dev
# ./MAKEDEV snd0 snd1 pcaudio
```

您可以使用下列的指令來聽 CD 音樂，不過要先將音樂 CD 放入光碟機喔：

```
# cdcontrol -f /dev/acd0c play
```

這裡我們假設您的光碟機代號為 acd0c，您如果不知道光碟機代號，可以使用 dmesg 來找出 cdrom 的裝置節點名稱。如果要停止播放，則使用下列指令：

```
# cdcontrol -f /dev/acd0c stop
```

3.3 編譯與安裝

3.3.1 編譯新的核心

修改完核心之後，接著要來編譯了。編譯核心有二種方式，第一種是使用傳統的編譯方式，使用下列指令，請將 MYKERNEL 改成您的核心檔名：

```
# config MYKERNEL
# cd ../../compile/MYKERNEL/
# make depend
# make
```

上面的指令完成後，應該就完成了核心的編譯。如果在過程中出現錯誤的訊息，注意一下錯誤是什麼，並再次修改。一般常見的錯誤就是網路卡驅動程式要求 miibus，而卻將它移除了。完成了上述的步驟確定沒問題時，就要安裝新的核心了。使用下列指令加以安裝

```
# make install
```

在 FreeBSD 4.2-STABLE 2000年2月以後的版本，我們可以使用新的方式來編譯及安裝新的核心：

```
# cd /usr/src
# make buildkernel KERNCONF=MYKERNEL
# make installkernel KERNCONF=MYKERNEL
```

安裝完成後，將編譯過程中使用的檔案刪除，並重新開機。

```
# cd / ; rm -rf /usr/src/sys/compile/MYKERNEL
# sync;sync;sync;reboot
```

如果您使用新的方式安裝核心，編譯過程所產生的檔案會放在 /usr/obj/usr/src/sys 中，請使用下列指令刪除：

```
# rm -rf /usr/obj/usr/src/sys/MYKERNEL
```

3.3.2 新的核心有問題

萬一很不幸的，你發現編完核心後，無法開機進入 FreeBSD，這時候就要使用舊的核心來開機了。在開機時看到倒數計時的時候，按 Enter 以外的鍵，會出現 boot: 這時候就先打 unload 來將已載入的資料移除，再打 /kernel.old 以使用舊的核心。

```
boot:  unload
boot:  /kernel.old
```

萬一連舊的核心也無法使用時，就使用安裝時的核心。

```
boot: /kernel.GENERIC
```

如果你想要刪除壞的核心，由於 kernel 檔有特殊的檔案屬性，因此必須先使用下列指令修改屬性，之後才可以刪除。

```
# chflags noschg /kernel
```

chapter 4

建立友善的界面

4.1 使用者介面設定

4.1.1 為什麼要更改設定

FreeBSD 安裝好後，我們必須做一些設定讓 FreeBSD 使用起來更順手。在預設的況狀下，系統並不支援中文檔名，對於使用中文的人不太方便。另外，命令列的樣子也不符合我們的需求。如果你想要自製化一些指令，也可以在這裡加上去。總而言之，在這個部份我們要做到的功能有下列幾點：

- 加上一些設定來支援中文的檔名
- 修改命令列指示，加上路徑名稱
- 更改預設的文書處理軟體及指令
- 建立中文終端機環境

FreeBSD 預設是使用 tcsh， 所以我們的設定都是以 tcsh 為主。

4.1.2 csh.cshrc

首先我們要將命令列改成下面這樣：

```
mydomain [/home/john] -john->
```

就是開頭是機器名稱，再來是路徑名稱，最後是使用者名稱。然後我們要將指令 ls 變成 ls -F，就是每次打 ls 時，系統會出現 ls -F 的效果。最後，因為vi 文書編輯器對於初學者而言不太好用，所以我們設定文書編

輯使用 ee。要這麼做必須編輯 /etc/csh.cshrc 及 ~/.cshrc。

首先編輯 /etc/csh.cshrc 加入下列設定：

```
setenv EDITOR ee
alias ls ls -F
set prompt = "%B%m [%/] -%n-> "
```

由於在 /etc 下的設定是通用的設定，但如果使用者自己的設定和通用設定一樣時，會以使用者的設定爲主。所以還要修改自己的設定，要這樣必須先編輯 ~/.cshrc，並找到setenv EDITOR vi 換成 setenv EDITOR ee，並加入一行 alias ls ls -F 才能有效喔。

如果希望日後每個使用者的設定都是這樣，必須修改 /usr/share/skel/dot.cshrc 檔案，該檔內容的修改和修改 ~.cshrc 一樣，把 vi 換成 ee，並加入 alias ls ls -F。因爲在使用 adduser 指令新增使用者時，它會問你是否要將 /usr/share/skel/dot.file 複製到使用者目錄下，因此我們就修改這裡，讓日後新增使用者時能使用該設定。

4.1.3 csh.login

接著要設定支援中文的 console 環境。編輯 /etc/csh.login 並在最後面加入以下的設定：

```
setenv ENABLE_STARTUP_LOCALE zh_TW.Big5
#若是使用遠端登入時,才能打出中文
setenv LC_CTYPE is_IS.ISO_8859-1
#若是console下用,才能打出中文
setenv LANG zh_TW.Big5
```

4.1.4 使用中文終端機

如果要在 console 下(就是在機器前面)使用中文的環境，必須安裝 big5con 軟體才可以。怎麼安裝呢？您可以將本書光碟二 /ports/distfiles 目錄中的所有檔案複製到 /usr/ports/distfiles中，或是將光碟放入光碟機中，並使用指令 mount /cdrom 將光碟機 mount 進來。如此一來，在安裝軟體時，系統就不會要求連上網路取回所需檔案(有的檔案不能從CDROM 安裝，您必須至網路下載)。接著以root 的身份直接使用下列指令來安裝：

```
# cd /usr/ports/chinese/big5con
# make install clean
```

安裝完後，使用指令 et 就可以出現像 DOS 下倚天中文的環境，如圖4-1。

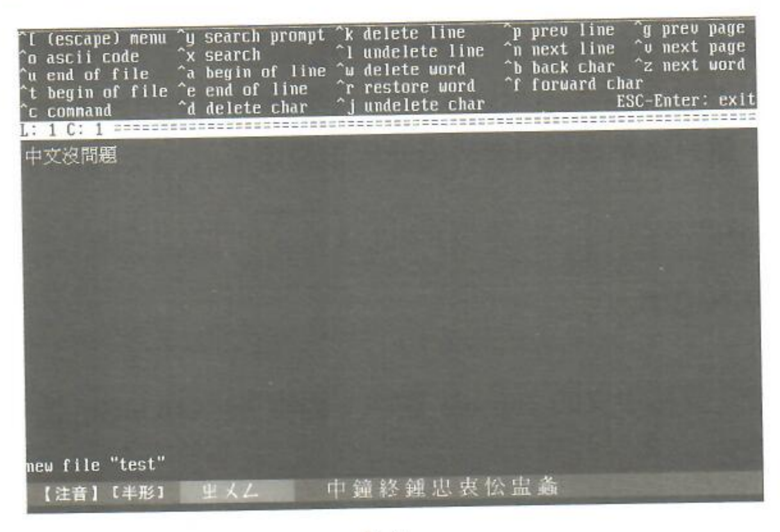


圖 4-1

4.2 登入前後的訊息修改

4.2.1 登入後的訊息

系統登入後，會自動秀出一段文字，稱爲 Message Of The Day(motd)。這一段文字是可以修改的，你可以編輯 /etc/motd 來製作自己的畫面。如果你想使用像 BBS 中的文書編輯軟體，來畫 ANSI 圖的話，你可以安裝 ve 這個軟體。

```
# cd /usr/chinese/ve
# make install clean
```

再使用 ve /etc/motd 來修改訊息。

4.2.2 登入前的訊息

在 telnet 到 FreeBSD 時，你會看到下面的畫面：

```
FreeBSD/i386 (alexwang.com) (ttyp0)
login:
```

我們在這裡要做的就是把它改成想要的樣子。更改 login 前的畫面有二種方式，一種是修改 /etc/gettytab 及 /etc/issue，另一種方式是修改 telnetd。

方式一：

編輯 /etc/gettytab，找到 default的地方。

```
default:\
:cb:ce:ck:lc:fd#1000:im=\r\n%s/%m (%h) (%t)\r\n\r\n:sp#1200:\
:if=/etc/issue:
```

其中的%s %m %h %t 分別對應到 FreeBSD i386 alexwang.com ttyp0，如果你不想顯示 FreeBSD ，就把 %s 拿掉。最後一行 if=/etc/issue 就是表如果沒有 issue 這個檔的話，就執行 default。

如果你不僅僅是要修改 FreeBSD/i386 這個部份，還想要在 login 前秀出一段文字的話，你可以新增 /etc/issue 這個檔，並編輯你想要的畫面。和 motd 一樣，issue 也可以使用 ANSI 畫面，所以你可以用 ve 來編輯畫面。如果你在該檔中加入 %s %m %h %t 等參數的話，也是會出現 FreeBSD i386 alexwang.com ttyp0等，如果不加就不會出現。

不過在 FreeBSD 4.5-RELEASE中，必須重新編譯 telnetd 才能使用 issue 作爲登入前畫面。所以要先執行下列指令：

```
# cd /usr/src/libexec/telnetd/
# make all install
```

方式二：

如果你想要在登入前執行一些指令，例如秀出開機時間等，必須要以更改 telnetd 的方式來做。編輯一個可執行檔 /usr/local/libexec/telnetd.sh 內容如下：

```
#!/bin/sh
TTY=`/usr/bin/tty | /usr/bin/cut -c9`

if [ "$TTY" = 'v' ]; then
```

```
exec /usr/libexec/telnetd
else
/bin/cat /etc/issue
echo "顏色控制碼`/usr/bin/uptime`控制結束碼"
exec /usr/libexec/telnetd
fi
```

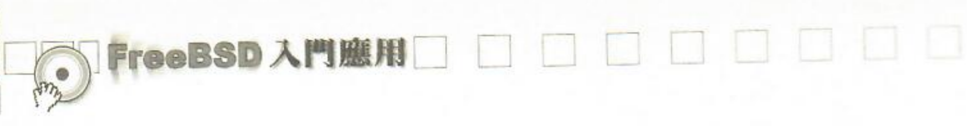
上面控制碼部份，你可以使用 ve 來加入顏色控制碼，編輯完後要把該檔變成可執行的檔案，使用下列指令：

```
# chmod 755 /usr/local/libexec/telnetd.sh
```

再編輯 /etc/inetd.conf，將原來的 "/usr/libexec/telnetd telnetd" 換成 "/usr/local/libexec/telnetd.sh telnetd.sh"：

```
telnet stream tcp nowait root  /usr/libexec/telnetd.sh telnetd.sh
```

最後重跑 inetd，使用指令 kill -HUP `cat /var/run/inetd.pid` 即完成設定。



chapter 5 使用者管理

5.1 帳號管理

5.1.1 新增使用者

新增使用者時，我們會以 adduser 這個指令來進行，adduser 指令其實是將新增使用者所必須做的事以 perl 寫成一個執行檔來自動幫我們做。為了了解系統對於使用者管理細節，首先，讓我們來複習一下如何使用指令 adduser 來新增使用者。當執行 adduser 指令後，系統會問我們一些問題：

```
輸入使用者的shell，我使用比較好用的 tcsh，故輸入 tcsh
Enter your default shell: csh date no sh tcsh [tcsh]:
使用者目錄要放在哪
Enter your default HOME partition: [/home]:
是否要從skel目錄中複製使用者的預設設定檔，直接按 Enter
Copy dotfiles from: /usr/share/skel no [/usr/share/skel]:
是否要送出訊息給使用者，直接按Enter，送出 /etc/adduser.message
Send message from file: /etc/adduser.message no  [/etc/adduser.message]:
是否要使用密碼，按 Enter ，預設是 y
Use passwords (y/n) [y]:
使用者的帳號，只能使用小寫的英文字母及數字
Enter username [a-z0-9_-]:
使用者全名，輸入你的真實姓名
Enter full name []:
要使用的 shell，打 tcsh
Enter shell csh date no sh tcsh [tcsh]:
使用者的目錄，按 Enter 即可
Enter home directory (full path) [/home/asdf]:
```

```
使用者的編號，按 Enter 即可
Uid [1001]:
登入的等級，按 Enter 即可
Enter login class: default []:
登入的群組，按 Enter 即可
Login group asdf [asdf]:
是否要該使用者加入其他群組，請注意，因為第一個使用者是你自己，所以在這裡要
打 wheel ，這樣你才可以 su 成 root ，也就是你的帳號才可以登入成 root 帳號
Login group is ``asdf''. Invite asdf into other groups: guest no [no]:
密碼
Enter password []:
再輸入一次
Enter password again []:
以上都正確嗎？
OK? (y/n) [y]:
```

最後還有一些問題，都是直接按 Enter 就可以了，等到它問我們是否要再增加下一個使用者時，可以答 y 來增加下一個使用者，也可以答 n 回到命令列。

看了 adduser 指令的過程，您對於新增使用者應有的步驟應該已經有初步的了解了，接下來我們要介紹 adduser 這個指令到底做了哪些事。

- **在 /etc/group 中加入使用者的群組**
- **在 /etc/master.passwd 中加入使用者**
- **在 /home 中建立使用者目錄，並建立 dotfile**
- **在 /var/mail 中建立使用者郵件目錄**
- **送出訊息給使用者**

知道了以上的流程，我們也可以自己做上述的步驟，但我們必須先知道 group 及 master.passwd 等檔案的格式。所以我們接下來要介紹這些檔案。

5.1.2 /etc/group介紹

在使用者的管理方面，FreeBSD 大致上可以分為群組管理及帳號管理。每一個帳號至少屬於一個群組，這樣子有利於權限控制。例如學生的帳號就有一個學生群組，而老師的帳號就屬於教師群組，某幾位老師屬於管理者的群組。這樣一來，我們除了可以針對個人設定權限外，也可以針對不同群組給予不同的權限。

/etc/group 這個檔案就是記錄群組的檔案，這是一個文字檔，我們可以使用 ee 等文字編輯軟體打開它。在 group 檔案中，其格式如下：

```
wheel:*:0:root,alex
students:*:1000:
teachers:*:1001:
```

每一個欄位以冒號分開，以最後一行為例，第一個欄位代表群組名稱為 teachers，而群組代號(gid)是 1001。我們可以自行使用文字編輯器加入想要的群組名稱及 gid，但要注意的是群組名稱和 gid 不能重覆。

第一行的最後面是 root,alex，在 FreeBSD 中，如果其他使用者要能使用 su 變成超級使用者的話，必須將其帳號加入 wheel 群組中。除了使用文字編輯外，也可以使用指令 pw 來新增群組：

```
# pw groupadd newgroup
# pw groupshow newgroup
newgroup:*:1002:
```

第一個指令是以參數 groupadd 來新增群組 newgroup，再以參數 groupshow 來顯示 newgoup 的資訊。

5.1.3 /etc/master.passwd介紹

FreeBSD 使用 shadow password 的方式來保護密碼檔，只有 root 才可以讀取編碼後的密碼檔 /etc/master.passwd。但是這並不是系統用來驗証的檔案，為了加快速度，FreeBSD 將該檔做成資料庫 /etc/spwd.db 及 /etc/pwd.db， 因此在修改完 master.passwd 後，必須使用指令 pwd_mkdb 來將 master.passwd 做成資料庫檔案。不過一般而言，我會使用 vipw 這個指令來修改master.passwd，vipw 會先將 master.passwd 以預設的文書編輯軟體打開，修改完存檔後，它會視需要自動更新資料庫。

執行 vipw 後，會出現：

```
root:$1$K9ooyz5O$b3D.jTTe.IWiwQMxo1Uft/:0:0::0:0:Charlie &:/root:/bin/csh
nobody:*:65534:65534::0:0:Unprivileged user:/nonexistent:/sbin/nologin
tom:Bk5AI4MiRKCJ2:1000:1000::0:0:Tom Chang:/home/tom:/bin/tcsh
```

它的格式是：name:password:UID:GID:class:change:expire:fullname:home:shell

- name：使用者帳號名稱，最多可以使用8個字元，不可重覆。
- password：可以是空的，代表不用密碼就可以登入，這樣很危險；也可以是 *，表示不可以登入；上面 vipw 顯示出來的項目中，以使用者 root 而言，他的密碼是使用 MD5 編碼過的，特徵是開頭為 $1 且看起來比較長；而使用者 tom 的密碼是使用 DES 編碼過的，DES 會將密碼編成一串13個字元的符號。
- UID：使用者代號，每個使用者都不一樣，不可重覆，如果有多個帳號使用同樣的 UID，FreeBSD 會將它當成同一個帳號。編號從 0 到 65535。UID 0 為系統中超級使用者的代號，內定只有 root 和 toor 的 UID 為 0。toor 帳號是 bash 所建立的使用者，內定不能使用該使用者登入。
- GID：群組代號，編號從 0 到 65535。
- class：除了群組外，class是更有彈性的控制方法，可以針對 /etc/login.conf 中不同的使用者設定來調整每個使用者的可使用的資源設定。
- change：強迫使用者變更密碼的時間，以從1970年到所要變更日期所經過的秒數來表示。你可以使用 date +%s 來求出從1970年到現在時間所經過的秒數，每天為86400秒，以現在時間的秒數加上86400*天數即為你要設定的時間。

  你可以使用指令

  ```
  expr `date +%s` + 86400 \* 30
  ```

  來取得30天後的秒數，再將其填入即可。若設為0則表示不使用此功能。
- expire：帳號的有效日期，一樣是以從1970年到到期日所經過的秒數來代表。若設為0則表示不使用此功能。
- fullname：使用者全名，你可以在此鍵入真實姓名。
- home：使用者的家目錄，即使用者登入後的所在目錄。
- shell：使用者的 shell。如果使用 /sbin/nologin 表示該名使用者不可以登入。

5.1.4 刪除使用者

知道了新增使用者的步驟後，您大概已經知道要怎麼刪除使用者了吧。只要把新增使用者的步驟反過來即可。

- 先移除使用者目錄 /home/tom，使用指令 rm -rf /home/tom
- 再移除使用者郵件目錄 /var/mail/tom
- 再移除使用者 crontab 檔案 /var/cron/tom
- 再以 vipw 來移除使用者帳號
- 若該群組已無人使用，則編輯 /etc/group 來移除群組。

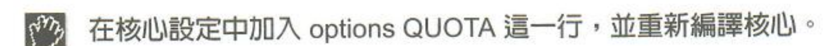
更簡單的方法是直接使用 rmuser 指令來移除，如 rmuser tom。使用 rmuser 可以快速的移除使用者，其執行結果如下：

```
Matching password entry:
tom:J7kPK0pKTn2oQ:1001:1001::0:0:Tom:/home/tom:/bin/tcsh
Is this the entry you wish to remove? y _真的要移除這一個使用者嗎？
Remove user's home directory (/home/test)? y _是否要移除使用者家目錄？
# 移除使用者正在執行的程式及更新系統資料
Removing user's at jobs:Updating password file, updating databases, done.
# 移除使用者群組
Updating group file: (removing group test -- personal group is empty) done.
# 移除使用者家目錄
Removing user's home directory (/home/test): done.
# 移除使用者信件
Removing user's incoming mail file /var/mail/test: done.
# 移除使用者暫存目錄
Removing files belonging to test from /tmp: done.
```

```
Removing files belonging to test from /var/tmp: done.
Removing files belonging to test from /var/tmp/vi.recover: done.
```

5.2 磁碟配額

當系統中有多位使用者時，如果其中一個擁有大量檔案，那麼其它使用者便無法有足夠的空間來使用。如果系統有許多使用者，而又不限制他們對磁碟的使用量，那麼磁碟很容易就會爆掉，因此我們必須對使用者加以限制。您可以限制系統中每個使用者可使用的硬碟大小(quota)。步驟如下：



- **在核心設定中加入** options QUOTA **這一行，並重新編譯核心。**
- **在** /etc/rc.conf **中加入一行** enable_quotas="YES"**。**
- **在** /etc/fstab **中要啓加磁碟限制的分割區中加入參數** userquota **，您也可以加入** groupquota **來限制群組的配額。**

```
# Device        Mountpoint   FStype   Options          Dump   Pass#
/dev/ad0s1b     none         swap     sw               0      0
/dev/ad0s1a     /            ufs      rw               1      1
/dev/ad1s1f     /home        ufs      rw,userquota     2      2
/dev/ad0s1e     /usr         ufs      rw               2      2
/dev/ad1s1e     /var         ufs      rw               2      2
```

- **重開機並設定使用者的** quota **限制。**

完成了上述的步驟並重新開機之後，我們就可以使用指令 edquota 來編輯磁碟配額。首先，以指令edquota -u tom 進入文書編輯，我們通常會加入二行，一行限制檔案大小，一行限制檔案數：

```
Quotas for user tom:
/home: blocks in use: 65, limits (soft = 50000, hard =55000)
inodes in use: 7, limits (soft = 5000, hard = 6000)
```

共中 blocks 代表使用者使用檔案大小總合，而 inodes 代表檔案數目。soft limits 代表使用量到多少時提出警告，而 hard limits 代表使用量達多少時立刻禁止寫入。

上面的範例中，使用者 tom 目前使用的檔案大小為 65K，在檔案大小總合為 50000K 時提出警告，55000K 時禁止寫入。目前該使用者有7個檔案，在檔案數達 5000個時提出警告，達6000個時禁止寫入。

我們也可以使用 edquota -p tom 2000-3000 來以使用者 tom 的設定為範例，將 UID 為 2000 到 3000 的使用者設定為和 tom 一樣。或者使用指令 edquota -p tom jack rose 來以使用者 tom 為範例，將 jack 及 rose 的設定變成和 tom 一樣。

您可以使用 quota -v tom 來看使用者 tom 目前的使用情形，或使用 repquota 來查看所有使用者目前的使用情形。

在 4.5-RELEASE 中，開機內定會檢查所有使用者的 quota，但這必須要花上一段時間，如果您不想在開機時自動檢查 quota 的話，請在 /etc/rc.conf 中加入 check_quotas="NO"。在 FreeBSD 3.2 版以前，開機內定是不檢查 quota 的，如果你想在開機時即檢查的話，請在 /etc/rc.conf 中加入 check_quotas="YES"

5.3 大量新增帳號

對於大型主機的管理者而言，要大量新增帳號時，若沒有一套"撇步"的話，使用 adduser 指令來新增帳號將會非常累人，所以我們必須要想出一個大量新增帳號的方法。

大量新增帳號的方法有很多，最簡單的就是寫一個程式來處理，只要依照之前提及的新增帳號步驟使用程式來一步步建立或在程式中呼叫 FreeBSD 系統中新增帳號的指令並經由迴圈來完成。但是 adduser 指令會問一堆問題，不適合拿來作程式中要呼叫的指令，所幸在 FreeBSD 中還有一個管理使用者帳號及群組的程式 pw，所以筆者就以指令 pw 加上一些控制來寫成 script。

一般而言，如果要新增的帳號是沒有規則性的，那麼就必須先將帳號做成一個文字檔，再以程式讀入。若帳號是有規則性的，則可以給定參數來完成。您可以在本書所附的第二片光碟中的 examples 目錄中找到筆者所寫的「新增大量帳號」程式，檔名是 adduser.tar.gz。

這個程式可以使用檔案匯入帳號及密碼，或只給帳號並自動產生密碼，最後將帳號及密碼存成一個檔案(預設是 adduser.log)。我們也可以使用連續產生帳號的方式，產生一堆連續的帳號，產生的結果同樣會放成 adduser.log 中。

首先，將第二片光碟放入光碟機中，並使用下列指令將檔案複製到 root 的目錄下：

```
# mount /cdrom
# cp /cdrom/examples/adduser.tar.gz /root/
```

將檔案放到硬碟中後，請切換目錄到 /root 並以下列指令解壓縮：

```
# cd /root
# tar zxvf adduser.tar.gz
```

解壓縮後，進入 adduser 目錄，再將 adduser.pl 更改權限為可以執行

```
# chmod 700 adduser.pl
```

adduser.pl 這個程式的使用方式如下：

表7

adduser.pl -f users.txt	參數 -f 或 -file 表示使用檔案匯入，檔案名稱可以自由命名，檔案每一行內容即一個帳號 的設定，每一行的格式可以是只有帳號或者是有帳號及密碼，帳號和密碼間使用逗點","隔開。本範例中將以檔案 user.txt 來產生帳號。
adduser.pl -u alex 001 003	參數 -u 或 -user 表示使用連續帳號，此範例將產生 alex001 alex002 alex003 三個帳號。

5.4 備份與移轉

在了解了新增使用者的步驟後，您對於備份使用者的作法在心中應該也有個譜了吧。在更新系統時，使用者的資料需要備份的有：

- /etc/master.passwd
- /etc/group
- 使用者目錄 /home
- 使用者郵件目錄 /var/mail
- 使用者定時執行的檔案 /var/cron/tabs

5.4.1備份

除非我們和使用者間已有共識， 不幫使用者備份其郵件及檔案，否則平常想要備份使用者資料的話，這是一件麻煩且費時的工作。不管是外在因素或是硬體固障，系統都有可能資料流失。對於一個公眾伺服器的管理者而言，事前明白告知使用者系統管理的原則是一件十分重要的事。如果未事前請使用者自行備份個人的檔案及郵件的話，首先，備份的工作將非常耗時，尤其是使用者擁有大量檔案時。再者，若未備份檔案，當系統資料流失時，容易和使用者產生爭議，就算平常每天都有備份，也只能保住備份當時的檔案，從備份到系統出問題的時候所產生的檔案就無法回復了。

如果不必備份使用者個人資料的話，就只需把 /etc/master.passwd 及 /etc/group 存在別的儲存設備或電腦中，要回復時只要依下列移轉的步驟做即可。

5.4.2 移轉

如果系統中已有其它使用者，要先編輯 /etc/group，加入和備份的 group 檔案有差異的地方，再使用 vipw 來將加入和備份的 master.passwd 有差異的地方。如果新系統中無其它使用者，則將所備份的 master.passwd 及 group 放到新電腦的 /etc 下，再執行下列指令以將密碼檔轉成資料庫格式即可：

```
# pwd_mkdb -p -d /etc /etc/master.passwd
```

如果沒有要移轉使用者個人資料的話，也必須建立使用者家目錄及郵件

目錄。如果要移轉使用者目錄的話，記得移轉後要檢查一下該目錄的所有人是不是該使用者。如果不是，就必須使用下列指令來將使用者目錄擁有者更改為所屬的使用者：

```
# chown -R user.group /home/user
```

上面這個指令是將 /home/user 這個目錄及其下所有目錄的所有人變成使用者名稱為 user，群組為 group。

我們也可以在密碼檔及群組資料移轉後，將舊的硬碟存放使用者資料的磁區(假設是 /home) mount 到 /mnt 下，再到 /mnt 下存放使用者資料的目錄中使用指令

```
# tar clf - . | tar xvpf - -C /home
```

來將使用者資料複製到 /home 中。並依此方法 mount 使用者郵件目錄磁區並複製到 /var/mail 下。

5.5 使用歷程記錄

5.5.1 記錄使用者指令

為什麼要記錄使用者下過的指令？並不是為了要監視使用者，而是在系統有問題時，可以找出原因。找出使用者曾經使用過哪些指令，以了解問題的所在。FreeBSD 提供指令 lastcomm 讓你查看系統中使用者執行的指令，但是你必須先修改 /etc/rc.conf，加上下列一行設定：

```
accounting_enable="YES"
```

系統會將使用者的歷程記錄在 /var/account/acct* 中，最新的記錄是 acct，隨著資料越來越多，系統每天會將該檔案移成 acct1，而 acc1 將變成 acct2，依此類推。如果站上使用者很多的話，acct 的檔案將變得非常大，所以你必須確保該目錄有足夠的空間存放資料，爲了避免檔案過多，系統內定只會保留最近四天的資料。

當下達指令 lastcomm 時，如果我們未使用任何參數，則系統會以 /var/account/acct 爲參考，印出所記錄的資料。你也可以使用 lastcomm -f acct1 來查看前一天的資料。

5.5.2 監看使用者

當使用者登入系統後，root 可以使用 watch 指令來取得線上使用者的視窗畫面。也就是說當下達指令後，root 所看到的畫面就會和該線上使用者一樣。我們可以觀察該使用者在做些什麼事，輸出的結果又是什麼。

這個指令並不是要給人拿來做壞事用的。如果有不友善的使用者登入系統時，可以使用它來觀察該使用者的動作，並適時阻止。

只有超級使用者 root 可以執行 watch，且在執行前必須先在 kernel 中加入下列的設定並重新編譯核心：

```
pseudo-device snp
```

並使用下列指令新增 snoop device：

```
# cd /dev
# ./MAKEDEV snp0 snp1 snp2 snp3
```

接下來就可 watch 指令了。首先，先下指令 w 來看一下站上有哪些使用者。指令結果的第二個欄位部份，有使用者的 tty，例如 p0、v0 等，選定要監看的使用者後，使用 watch ttyp0 來監看該使用者，其中 ttyp0 即該使用者的 tty。你可以使用 CTRL+X 來切換不同的 tty，也可以使用 CTRL+G 離開回到自己的畫面。

5.5.3 控制 root 的使用

如果有在 wheel 群組中加入一般的使用者，則該使用者可以使用指令 su 並輸入 root 的密碼後變成超級使用者。但如果系統中有多位使用者具有 root 的權限，我們根本不知道是誰使用了 root 的權限、執行了哪些指令；如果我們想針對不同人給予不同的權限，例如一個人只有備份的權限、另一個人只能觀看系統設定，su 也無法達成我們的要求。因此有人發展出 sudo 這個軟體來支援系統的管理。

不過 sudo 並不是 FreeBSD 系統內定的指令，我們必須自己安裝。所幸 FreeBSD 已將該軟體移植到 "port" 中，我們只要執行下列指令即可輕鬆的安裝了。不過您必須先將所附光碟二 /ports/distfiles 目錄中的檔案複製到 /usr/ports/distfiles中，或者先將光碟 mount 進來，否則電腦必須先連上網路喔！由於在 port 中 sudo 的安裝設定並未打開 sudo 執行指令記錄，如果你希望它能將執行 sudo 的 log 記下來的話，必須先編輯 /usr/ports/security/sudo/Makefile，將 CONFIGURE_ARGS 中的參數 --disable-log-wrap 拿掉。

```
# cd /usr/ports/security/sudo
# make install clean
```

安裝完後，要先執行 /usr/local/sbin/visudo 以設定 sudo 的設定檔 (/usr/loca/etc/sudoers)。以下簡單說明該設定檔如何設定，詳細說明及範例請參考/usr/local/etc/sudoers.sample：

```
# Host alias specification(如果你只有一台機器，可以不必設)
Host_Alias CUNETS = 128.138.0.0/255.255.0.0
Host_Alias      SERVERS = master, mail, www, ns
# User alias specification(把使用者分成群組，也可以不用分)
User_Alias      PARTTIMERS = millert, mikef, dowdy
User_Alias      WEBMASTERS = will, wendy, wim
User_alias      BACKUP = tom, jack
# Cmnd alias specification(把可以執行的指令分成一個個群組)
Cmnd_Alias READ=/bin/ls,/bin/cat,/usr/bin/tail,/usr/bin/head
Cmnd_Alias BACKUP=/sbin/dump,/usr/bin/tar,/usr/bin/find, \
        /usr/bin/cpio,/bin/cp,/sbin/mount,/bin/dd
# User privilege specification(設定使用者的權限)
root    ALL=(ALL) ALL
%wheel          ALL = (ALL) ALL
# 表示 jack 在 CUNET 中所有機器都可執行指令群組中 BACKUP 指令
jack    CUNETS = BACKUP
# 表示使用者群組中 PARTTIMERS 群組的人，只能在 SERVERS
#群組中的機器裡執行 READ 群組的指令
PARTTIMERS SERVERS = READ
```

設定完後，使用者即可執行 sudo <允許的指令> ，使用者只要輸入自己的密碼即可，不必知道 root 的密碼，而且 5 分鐘內再次執行 sudo 時不需再輸入密碼。如果你有打開 log 記錄功能，sudo 執行成功或失敗的 log 都將被記錄到 /var/log/sudo.log 中。詳細說明請閱讀說明 man sudo。

chapter 6 網路設定

6.1 固接網路

如果您的網路連結方式是固接網路，如學校的宿網、固接式 ADSL、固接式 Cable，恭喜你，這種設定最簡單，而且你的連線速度應該令人羨慕。這樣的網路設定可以經由下列方式達成：

- 使用 /stand/sysinstall
- 手動設定

6.1.1 使用 /stand/sysinstall

以 root 身份，執行 /stand/sysinstall 進入安裝時的畫面。

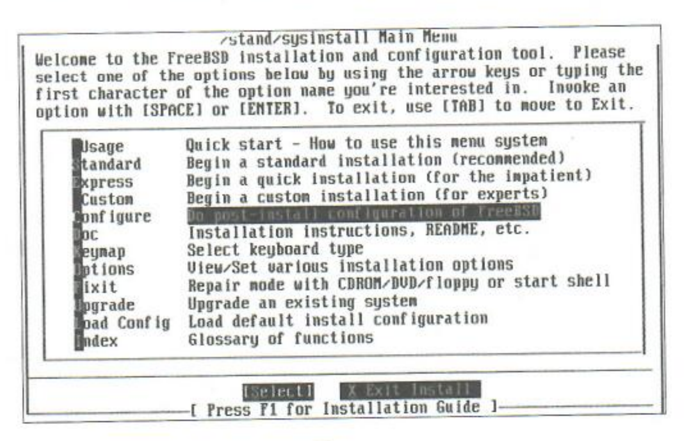
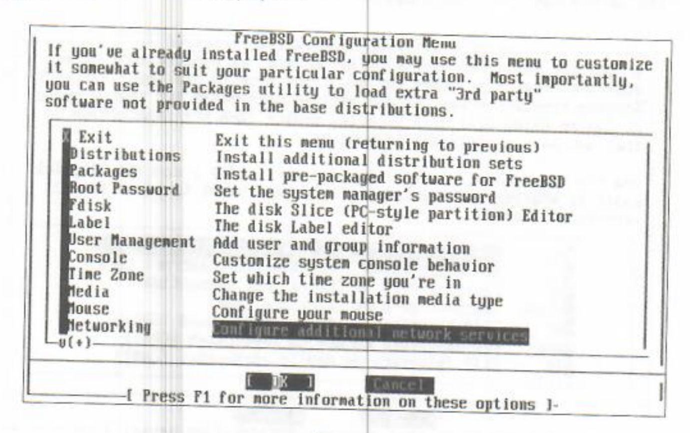


圖 6-1

選擇 Configure 進入，出現圖6-2：


FreeBSD Configuration Menu

If you've already installed FreeBSD, you may use this menu to customize it somewhat to suit your particular configuration. Most importantly, you can use the Packages utility to load extra "3rd party" software not provided in the base distributions.

X Exit	Exit this menu (returning to previous)
Distributions	Install additional distribution sets
Packages	Install pre-packaged software for FreeBSD
Root Password	Set the system manager's password
Fdisk	The disk Slice (PC-style partition) Editor
Label	The disk Label editor
User Management	Add user and group information
Console	Customize system console behavior
Time Zone	Set which time zone you're in
Media	Change the installation media type
Mouse	Configure your mouse
Networking	Configure additional network services

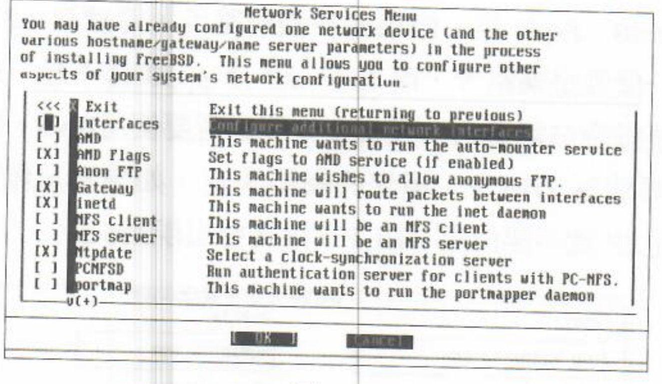
v(+)

[OK] Cancel

[Press F1 for more information on these options]

圖 6-2

接著選擇 Networking 進入，進入圖6-3：


Network Services Menu

You may have already configured one network device (and the other various hostname/gateway/name server parameters) in the process of installing FreeBSD. This menu allows you to configure other aspects of your system's network configuration.

<<<	X Exit	Exit this menu (returning to previous)
[ ]	Interfaces	Configure additional network interfaces
[ ]	AMD	This machine wants to run the auto-mounter service
[X]	AMD Flags	Set flags to AMD service (if enabled)
[ ]	Anon FTP	This machine wishes to allow anonymous FTP.
[X]	Gateway	This machine will route packets between interfaces
[X]	inetd	This machine wants to run the inet daemon
[ ]	NFS client	This machine will be an NFS client
[ ]	NFS server	This machine will be an NFS server
[X]	Ntpdate	Select a clock-synchronization server
[ ]	PCNFSD	Run authentication server for clients with PC-NFS.
[ ]	portmap	This machine wants to run the portmapper daemon

v(+)

[OK] Cancel

圖 6-3

接著選第一個 Interfaces，出現圖6-4：

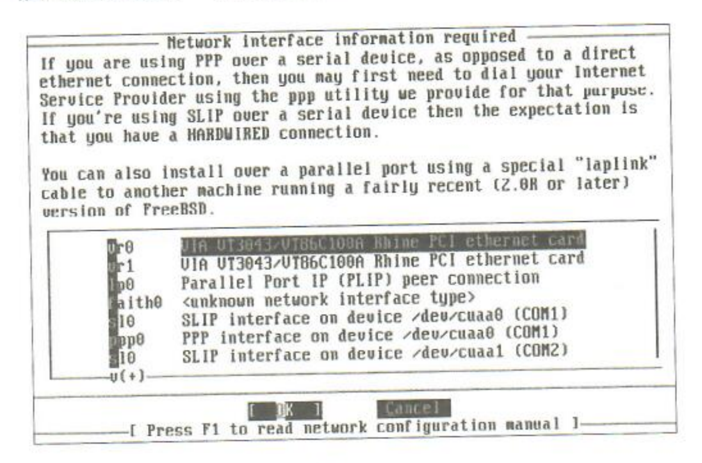


圖 6-4

上圖中的 vr0 即是您的網路卡，vr0 可能會因爲網路卡的不同而有不同的代號，如 ed0、fxp0 等。如果有多張網路卡，還會有 ed1、vr1、fxp1 等。通常第一個就是網路卡，而 lp0、sl0 及 ppp0 都不是。選擇您想要設定的網路卡後按空白鍵進入，程式會先問是否要使用IPv6，回答否。接著會問您是否要使用 DHCP，視您的網路決定，如果不是動態取得 IP 的話，擁有固定 IP 就不要使用 DHCP。接著會出現圖6-5：

Network Configuration
Host: www.mydomain.com
Domain: mydomain.com
IPv4 Gateway: 140.115.1.254
Name server: 140.115.1.31
Configuration for Interface vr0
IPv4 Address: 140.115.75.1
Netmask: 255.255.0.0
Extra options to ifconfig:
[OK]
CANCEL

圖 6-5

我們要先知道我們的 Domain Name 及 IP 等，如果沒有 Domain Name 的話，就隨便輸入吧。假設我們的機器 www.mydomain.com ，Host 一欄中就輸入 www.my-domain.com，在 Domain 中就輸入 mydomain.com。我的機器在中央大學，所以 Gateway 就輸入 140.115.1.254，Name server 輸入 140.115.1.31，IPv4 Address 就輸入我的 IP 也就是 140.115.75.2，Netmask 也就是子網路就輸入 255.255.0.0。接著按 OK 離開就完成了，它會問你是否用立即使用新的網路設定，回答是就會立即更新網路設定了，接著就可以離開程式了。

雖然 sysinstall 有問我們是否要立即使用新的網路設定，但它不一定會立即更新設定，所以我們需要重新開機或是使用指令來將網路立刻更新。至於如何不重開機而更換 IP 設定，我們在下面手動設定時將提及。

6.1.2 手動設定

手動設定比用 /stand/sysinstall 設定還要快且簡單。只要知道我們的網路卡代號、IP 、Netmask等資料就可以開始設定了。 您可以使用指令 dmesg | grep Ethernet 或是 ifconfig 去看網路卡代號爲何。首先，爲了在一開機即設定，必須在 /etc/rc.conf 中依你的資料加入下列幾行：

```
#Gateway
defaultrouter="140.115.1.254"
#Host，機器的Domain Name
hostname="www.ba.ncu.edu.tw"
#網路卡代號是 vr0，設定 IP為 140.115.75.2，子網路遮罩為 255.255.0.0
ifconfig_vr0="inet 140.115.75.2 netmask 255.255.0.0"
```

接著編輯 /etc/resolv.conf，依您的資料加入下列幾行：

```
#網域(domain)
domain ba.ncu.edu.tw
#DNS伺服器位址
nameserver 140.115.1.31
```

以上資料都設定好了之後就可以重新開機使用新的設定了。或者你也可以使用下列指令來更新 IP。下面的指令中，網路卡代號為 vr0，IP 是 140.115.75.2，子網路遮罩是 255.255.255.0。

```
# ifconfig vr0 down
# ifconfig vr0 140.115.75.2 netmask 255.255.255.0
# ifconfig vr0 up
```

6.2 ADSL

如果家裡有多台電腦要上網，使用 FreeBSD 來做連線分享是一件很棒的事。ADSL 的撥號是使用 PPPoE (PPP over Ethernet) 的方式，由於撥接式 ADSL 只有一個 IP，因此家中其他的電腦必須使用保留 IP 再經由 FreeBSD 的 NAT (Network Address Translation) 功能來將保留 IP 轉成可以在網際網路上出現的 IP。

在這一部份我們將說明如何將 FreeBSD 使用 ADSL 連上網路，並擁有 NAT 功能。如果您只有 FreeBSD 要上網，你只需要使用一張網路卡，再加上下列關於 PPPoE 的設定即可。如果你有多台電腦要經由 FreeBSD 上網，除了 ADSL 的設定外，還要再加上 NAT 的設定。你必須準備二張網路卡，一張連接到 ADSL Modem，另一張連接到區域網路的 Hub。



6.2.1 編譯核心

如果您使用的是 FreeBSD 4.4-Release 以後的版本，您不需要修改核心設定就可以支援 PPPoE 了，因爲當系統要求使用 PPPoE 時，會以動態的方式載入。如果使用的是 4.4 以前的版本，還是要加上關於 PPPoE 的設定。首先我們要先確定在核心中已經有加上網路卡的設定，也就是開機時已經有抓到網路卡了。接著請先在 kernel 中加入下列幾行：

```
#PPPoE 方面(FreeBSD 4.4-RELEASE 以後的版本不需加入下列三行)
options NETGRAPH
options NETGRAPH_PPPOE
options NETGRAPH_SOCKET

#NAT 方面(如果不使用NAT可以不加)
options IPFIREWALL
options IPDIVERT
options IPFIREWALL_DEFAULT_TO_ACCEPT
```

接著要重新編譯核心

```
# config KERNEL
# cd ../../compile/KERNEL
# make depend
# make
# make install
```

6.2.2 修改 /etc/ppp/ppp.conf

這裡我們以 Hinet 的 ADSL 爲範例說明，其他家的 ADSL 設定大都差不多。首先，將 /etc/ppp/ppp.conf 更名爲 /etc/ppp/ppp.conf.old：

```
# mv /etc/ppp/ppp.conf /etc/ppp/ppp.conf.old
```

再來，使用文書編輯軟體來新增並編輯 /etc/ppp/ppp.conf，加入下列的設定，請記得要修改下列設定中的 "set device PPPoE:---" 那一行，在那一行設定連接到 ADSL 的網路卡代號，範例中是使用 vr0 。接著要修改 authname 及 authkey 成爲你的帳號及密碼。

```
# /etc/ppp/ppp.conf
default:
  set log Phase Chat LCP IPCP CCP tun command
  nat enable yes
  nat same_ports yes
  nat use_sockets yes
  set redial 15 28800
  set reconnect 15 28800

pppoe:
  set device PPPoE:vr0:
  set mru 1492
  set mtu 1492
  set speed sync
  enable lqr
  set lqrperiod 5
```

```
  set cd 5
  set dial
  set login
  set timeout 0
  set authname b1xxxxxx@hinet.net
  set authkey yourpassword
  set ifaddr 10.0.0.1/0 10.0.0.2/0 255.255.255.0 0.0.0.0
  add default HISADDR
  enable dns
# end of ppp configuration
```

完成後即可存檔離開。

6.2.3 修改 /etc/rc.conf

再來是修改 /etc/rc.conf，在這裡我們一樣是以 vr0 為連接到 ADSL 的網路卡代號，以 vr1 為連接到區域網路的網路卡代號(如果沒有要使用 NAT 則可以不必設定)，請記得要修改成你的網路卡代號。 然後請在 /etc/rc.conf 中加入下列幾行：

```
# /etc/rc.conf
#設定自動選擇連線裝置
network_interfaces="auto"
ifconfig_vr0="inet 10.0.0.1 netmask 255.0.0.0 -arp up"
#一開機就執行 PPPoE(建議)
ppp_enable="YES"
#ddial 表示只要斷線便自動連線(建議)，或設 auto 表示有資料要
```

```
#出去則自動連線，也可以設為background只連一次後放到背景
ppp_mode="ddial"
ppp_profile="pppoe"
defaultrouter="10.0.0.1"
gateway_enable="YES"

#以下的設定如果不使用 NAT 則可省略
ifconfig_vr1="inet 192.168.0.1  netmask 255.255.255.0"
firewall_enable="YES"
firewall_type="OPEN"
natd_interface="vr0"
natd_enable="YES"

#end of  /etc/rc.conf
```

完成了上述步驟之後，就可以重新開機以啓動 PPPoE了。如果您在 /etc/rc.conf 中並未設定一開機就自動連線的話，可以在重開機之後使用以下的指令來連上 internet：

```
# ppp -background pppoe
```

6.2.4 分享網路連線

如果我們要將 FreeBSD 的網路連線分享給家中其他電腦使用，除了上述的 NAT 設定外，在其他電腦還要再做一些設定。首先，我們的網路架構應該如圖 6-6 所示：

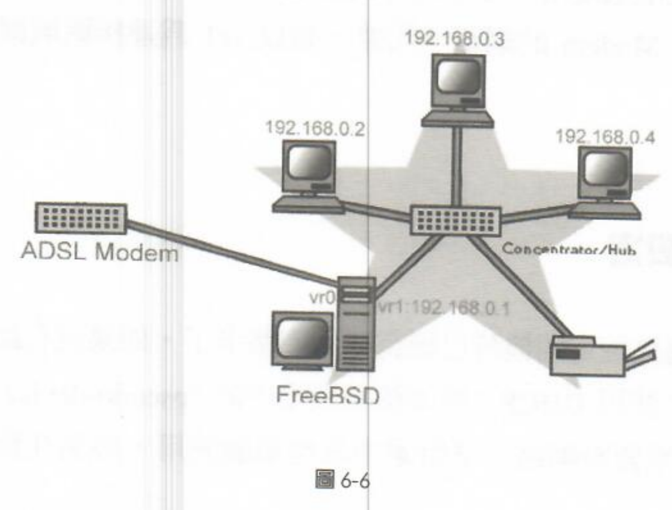


圖 6-6

在客戶端其他的電腦設定方面，我們必須將其他電腦的 IP 設定為 192.168.0.X、子網路遮罩是 255.255.255.0，gateway 設定為 FreeBSD 連到區域網路的網路卡 IP，在此範例中是 192.168.0.1。然後再設定 DNS 為您 ISP 的 DNS，以 Hinet 而言是 168.95.1.1。

完成上述的設定後，我們就能享受以 FreeBSD 為連線分享器快速上網了。

6.3 Cable Modem

如果你是使用 Cable Modem 來連上網路，你只需經由 DHCP 動態取得 IP 即可。同樣的，你也可以和 ADSL 一樣將 Cable Modem 的網路連線分享給家中其他電腦使用。分享的方式也是經由 NAT。在這裡我們將以 vr0 爲連向 Cable Modem 的網路卡代號，而以 vr1 爲連接區域網路的網卡代號。

6.3.1 核心設定

首先，你應該確定開機時已經有抓到網路卡了，如果沒有請重新編譯核心。而且爲了使用 DHCP，原本核心設定中的 "pseudo-device bpf" 不可以刪除喔。 如果要將網路分享給家中其他電腦使用，必須在核心中加入下列設定：

```
#NAT 方面(如果不使用NAT可以不加)
options IPFIREWALL
options IPDIVERT
options IPFIREWALL_DEFAULT_TO_ACCEPT
```

接著要重新編譯核心：

```
# config KERNEL
# cd ../../compile/KERNEL
# make depend
# make
# make install
```

接著要檢查一下 /dev 中是否有 bpf* 的檔案，如果沒有，請執行下列指令以建立：

```
# cd /dev
# ./MAKEDEV bpf0 bpf1 bpf2 bpf3
```

6.3.2 設定/etc/rc.conf

首先，將網路卡和 Cable Modem 連接好，假設網路卡代號是 vr0。我們先執行下列指令來連接到 internet：

```
# dhclient vr0
```

接著打 ifconfig vr0，你應該可以看到下列畫面

```
vr0: flags=8843 mtu 1500
inet6 fe80::250:baff:fe00:dcdd%vr0 prefixlen 64 scopeid 0x1
inet 61.58.76.14 netmask 0xffffff00 broadcast 61.58.76.255
```

上面畫面中的 61.58.76.14 是我們動態取得的 IP，如果有出現 IP 就表示正常了。然後試一下可不可以 ping 到外面的網路：

```
# ping 216.136.204.21
```

都沒問題後就可以開始編輯 /etc/rc.conf 了，請在 rc.conf 中加入下列設定：

```
# Cable Modem的設定
ifconfig_vr0="DHCP"
```

```
#以下的設定如果不使用 NAT 則可省略
ifconfig_vr1="inet 192.168.0.1  netmask 255.255.255.0"
gateway_enable="YES"
firewall_enable="YES"
firewall_type="OPEN"
natd_interface="vr0"
natd_enable="YES"
```

完成上面的步驟就可以重新開機使用 Cable Modem 上網了。

6.3.3 連線分享

如果你要將 FreeBSD 的網路連線分享給家中其他電腦使用，除了上述的 NAT 設定外，在其他電腦還要再做一些設定。首先，你的網路架構應該如圖6-7所示：

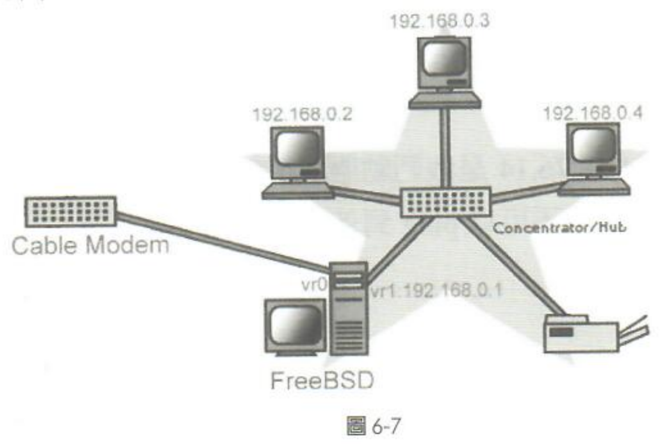


圖 6-7

接著，在其他的電腦設定方面，我們必須將其他電腦的 IP 設定為 192.168.0.X、子網路遮罩是 255.255.255.0，gateway 設定為 FreeBSD 連到

區域網路的網路卡 IP，在此範例中是 192.168.0.1。然後再設定 DNS 爲您 ISP 的 DNS。

完成上述的設定後，就能享受以 FreeBSD 爲連線分享器使用 Cable modem 快速上網了。

6.4 Modem 撥接

如果你是使用 modem 撥接上網的話，你同樣可以在 FreeBSD 中設定。只要是用需要撥接的連線方式，都是使用 PPP 來連接。而且就算使用 modem 撥接一樣可以分享給區域網路中其他電腦使用。

6.4.1 編輯 /etc/ppp/ppp.conf

這裡我們以 Hinet 爲例加以說明。首先，先將原本的 ppp.conf 更名爲 ppp.conf.old，再編輯 /etc/ppp/ppp.conf，如下：

```
default:
 set log phase chat connect LCP IPCP CCP tun command

 #設定使用哪一個 com，com1 是 cuaa0、com2 是 cuaa1
 set device /dev/cuaa1
 set speed 115200
 deny lqr
 set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 5 \"\" \
```

```
AT OK-AT-OK ATE1Q0 OK\\dATDT\\T TIMEOUT 40 CONNECT"

hinet:
 set openmode active
 #設定撥接的號碼
set phone 4125678

# 設定共撥3次,每次隔5秒
set redial 5 3

#設定閒置幾秒就自動斷線，0表示不會自動斷線
set timeout 1200

#設定可以使用 ppp 的系統使用者帳號
allow users xxx
deny chap
disable chap
accept pap

#Hinet 的撥接帳號 xxxx
set authname xxxx

#Hinet 的撥接密碼 xxxx
set authkey xxxx
set ifaddr 10.1.1.1/0 10.2.2.2/0
```

設定完後存檔離開進入下一個步驟。



6.4.2 編輯 /etc/ppp.linkup

接著編輯 /etc/ppp.linkup 如下：

```
hinet:
    delete ALL
    add 0 0 HISADDR
```

存檔離開，現在可以使用指令 ppp hinet 來撥接了。

6.4.3 網路分享

如果你希望將 modem 的連線分享給其他區網中的電腦使用，您必須要有一張網路卡連到區域網路中。並設定其 IP 為 192.168.0.1、子網路遮罩為 255.255.255.0。在區網中的其他電腦要將 gateway 設定為 192.168.0.1，IP就設定為 192.168.0.2，子網路遮罩為 255.255.255.0。接著使用指令來撥號，就可以了：

```
# ppp -nat hinet
```

6.5 網路相關指令

這裡我們將介紹一些常用到的網路相關指令，如果你想知道更多、更詳細的指令，請參考指令應用篇，或使用指令 man 來查詢相關使用方法。在這裡我們的目的只是告訴你有這些工具可以使用，並未針對每一個指

令作詳細的使用方法說明，你應該使用 man 來查詢該指令更完整的參數。我只列出較常用的幾個方法。

6.5.1 telnet

大家都知道這個指令吧！如果你要連線到別台 UNIX 主機或是連到 BBS，就使用 telnet 這個指令。例如，要telnet 到 sparc20.cc.ncu.edu.tw 這台機器：

```
$ telnet sparc20.cc.ncu.edu.tw
```

如果你在登入時想離開，可以按 CTRL+] 回到自己的主機，再打 quit 離開 telnet。如果你在 telnet 時想輸入中文的話，必須加入參數-8：

```
$ telnet -8 bbs.ba.mgt.ncu.edu.tw
```

6.5.2 ftp

FreeBSD 中也有提供命令列的 ftp 工具，如果你要連到 freebsd.csie.nctu.edu.tw 這台機器的話：

```
$ ftp freebsd.csie.ncu.edu.tw
```

允許匿登入的 ftp 主機，帳號只要輸入 anonymous 即可，密碼可以隨便輸入，或者在 ftp 指令之後加上參數 -a 即可。進入 ftp 站台後，你可以使用下列指令：

表8

!	列出所有指令。
ls	查看所在目錄的檔案，使用方法和在 FreeBSD 機器中一樣。
cd	進入某一個目錄，如 cd pub。
get <filename>	取回某一個檔案，如 get ve-1.0.tgz。
reget <filename>	續傳某一檔案。
put <filename>	上傳檔案，如 put homework01.zip。
send <filename>	上傳檔案，和 put 一樣。
size <filename>	查看檔案大小。
less <filename>	觀看文字檔內容

6.5.3 ping

送出 ICMP 封包，用以查看網路上主機的連線狀況。

```
$ ping 216.136.204.21
```

你也可以加入以下的參數：

-c count 只計算 count 次。

-s size　不使用預設的 64 bytes 當作封包大小，而改用新的 size。

```
$ ping -c 10 -s 108 216.136.204.21
```

6.5.4 nslookup

查詢網路主機資訊。此指令可以用來查詢網路主機的 Domain name，或以 Domain name 反查 IP 位址。

```
$ nslookup 216.136.204.21
```

```
$ nslookup www.freebsd.org
```

6.5.5 netstat

顯示網路狀況。可以用來看網路的組態及各項服務的情形。

參數：

-a	顯示所有資訊
-n	以 number 方式顯示 IP 位址
-i	顯示網路介面

```
$ netstat -a
```

```
$ netstat -ni
```

6.5.6 traceroute

追蹤網路路徑，用這個指令，你可以知道從你的主機到某一台主機的過程中經過了哪些機器。

使用方法：traceroute www.freebsd.org

6.5.7 sockstat

查看主機 internet 或 domain socket。你可以用來查詢有誰連到你的機器中，由哪一個網路服務接收，該網路服務的 PID 是多少等。還可以查詢本機開放了哪些 port、提供了哪些服務。

6.5.8 ifconfig

設定或檢查網路介面，ifconfig 可以用來設定你的網路卡，顯示網路介面的資訊。

參數：

-a	詳細顯示所有介面
-u	顯示目前使用中的裝置
interface	顯示該interface 的資訊，此 interface 為你的網路卡代號或其他代號。
down	停用某一裝置
up	啟用某一裝置
ifconfig -a	顯示所有介面的資訊
ifconfig vr0	顯示網路卡 vr0 的資訊
ifconfig vr0 down	停用網路卡 vr0
ifconfig vr0	192.168.0.1netmask 255.255.255.0 設定網路卡 vr0 的ip
ifconfig vr0	up 啟用網路卡 vr0

6.5.9 tcpdump

列出所有到達本機的 tcp 封包。

6.5.10 lynx

文字瀏覽器。這並不是 FreeBSD 內定的指令，所以必須先使用 port 安裝：

```
# cd /usr/ports/www/lynx
# make install clean
```

安裝完就可以使用 lynx 來上網瀏覽網頁

```
$ lynx www.freebsd.org
```

我們也可以使用 lynx 來下載網頁上的檔案。例如，我們要檔案位置是 http://www.apache.org/dist/httpd/apache_1.3.22.zip，可以使用下列指令來下載它，並存成 apache.zip 這個檔名：

```
$ lynx -dump http://www.apache.org/dist/httpd/apache_1.3.22.zip > apache.zip
```

chapter 7 /etc 目錄下的檔案介紹

/etc 是 FreeBSD 系統主要設定檔所在，了解這個目錄下的檔案及其格式，對於我們管理及使用 FreeBSD 將有更深入的認識。因此，以下我們就分別說明一下這些檔案。

7.1 aliases

用以告知 sendmail 要將信轉給哪個使用者或是交由哪個程式處理。請注意，修改完這個檔案後，必須使用指令 newaliases 來讓所做的修改在 sendmail 中發生作用。這個檔案是用來設定郵件的別名，也就是可以設定要將某人的信件轉給其他地方(人員或程式)。你也可以將某人的信轉給很多人，這個檔案的位置是由 sendmail.cf 檔案中的 AliasFile 這個選項所決定的。當 sendmail 收到信時，會一行行比對，當第一行符合後，就不會再繼續下去，所以應注意優先順序。

語法：開頭的 "#" 代表該行是註解，大小寫都視為一樣

```
1.root:          alex
2.webmanager:          alex,jack,jim@other.hostname.com
3.nobody:       /dev/null
4.homework:              |/usr/local/bin/homework.sh
5.olduser:      :include:          /usr/local/olduser_list
```

範例 1　是將寄給 root 的信轉給本機中的使用者 alex。

範例 2　是將寄給 webmanager 的信轉給本地的使用者 alex,jack及別地的jim@other.hostname.com。

範例 3　是將寄給 nobody 的信直接丟掉，丟入 /dev/null 這個無底深淵。

範例 4 是將寄給 homework 的信交給 /usr/local/bin/homework.sh 這支程式處理。

範例 5 是將寄給 olduser 的信轉給檔案 /usr/local/olduser_list中所列出的所有使用者。olduser_list 為使用者清單的文字檔。

當設定了一堆複雜的別名之後，我們要看最後信會寄到哪裡時，可以使用下面指令來看寄給 username 的信最後寄給誰：

```
# sendmail -bv username
```

aliases檔中將很多東西都轉向 root，因此你可以去讀 root 的信箱或是將 root 的信轉給別的地方，下面這一行是將 root 的信都轉給 my@my.domain：

```
root :          me@my.domain
```

當郵件無法送出被退回時給使用者時，都是以 MAILER-DEAMON 爲帳號寄出。因爲使用者可能會回覆那封被退回的信，所以這個別名是必備的。而 postmaster 則負責處理所有關於郵件問題的信件，因此也是必備的，一定要保留下面二行，這是必要的系統基本設定：

```
MAILER-DAEMON: postmaster
postmaster: root
```

7.2 crontab

讓系統管理者設定定期執行的工作。你也可以用 root 執行 crontab -e 來取代這個檔。

```
# 設定使用的 shell, 路徑
SHELL=/bin/sh
PATH=/etc:/bin:/sbin:/usr/bin:/usr/sbin
HOME=/var/log
#
# 分       小時   天     月      星期幾   身份   指令
#minute   hour   mday   month   wday     who    command
#
*/5       *      *      *       *        root   /usr/libexec/atrun
```

- minute：代表一小時內的第幾分，範圍 0-59
- hour：代表一天中的第幾小時，範圍 0-23
- mday：代表一個月中的第幾天，範圍 1-31
- month：代表一年中第幾個月，範圍 1-12
- wday：代表星期幾，範圍 0-7 (0及7都是星期天)
- who：要使用什麼身份執行該指令
- command：所要執行的指令

小時的欄位中如果是 *，表示每小時，天的欄位中如果是 *，表示每天，依此類推。欄位中也可以使用 "-" 來表示範圍，例如，在小時的欄位

中填 8-11，表示執行的時間是 8,9,10,11，共四次。

欄位中也可以用逗點來表示，以分的欄位而言， 1,2,5,9 表示將在 1,2,5,9 分時各執行一次。我們也可以寫成像這樣 1-2,12-14 ，表示在 1,2,12,13,14 分各執行一次。

另外，也可以用 / 後面加數字表示每幾分鐘要執行一次。如在分的欄位填 0-23/2，表示 1-22 分之間，每隔二分鐘執行一次，也就是 0,2,4,6,8,10,12,14,16,18,20,22。如果在分的欄位是 */5，表示每五分鐘一次。

除此之外，也可以用一個開頭為 @ 的字串來表示各種意義：

表9

字串	代表意義
@reboot	開機時跑一次。
@yearly	每年跑一次,等於 "0 0 1 1 *"。
@annually	和 @yearly 一樣。
@monthly	每月跑一次,等於 "0 0 1 * *"。
@weekly	每週跑一次,等於 "0 0 * * 0"。
@daily	每天跑一次,等於 "0 0 * * *"。
@midnight	和 @daily 一樣。
@hourly	每小時跑一次,等於 "0 * * * *"。

我們在安排 crontab 時，應該要錯開每個程式的執行時間，才不會有一大堆程式同時執行，造成系統負荷過高。

7.3 csh.cshrc

這是 Shell csh、tcsh 用的內定 .cshrc 檔案，也就是進入該 Shell 時會載入的設定。

```
# 檔案權限設定
umask 022

# 設定內定使用的文字編輯器為 ee
setenv  EDITOR  ee

# 設定當使用者打 ls 時，出來的結果是 ls -F 的結果
alias  ls     ls -F
# 設定當使用者打 cd.. 時，變成是打 cd ..
alias cd.. 'cd ..'

# 設定命令提示符號為 "主機名稱 [所在目錄] -使用者名稱->"
set prompt = "%B%m [%/] -%n-> "
```

7.4 csh.login

這是 Shell csh、tcsh 用的內定 .login 檔案，也就是進入該 Shell 時會載入的設定。

```
# 要讀取系統訊息則將下面一行的 # 拿掉
# msgs -f

# 允許終端機訊息，設為 y 才可以使用 write 的指令，傳送訊息給其他使用者
# mesg y

# 設定支援中文的終端機
setenv ENABLE_STARTUP_LOCALE zh_TW.Big5
setenv LC_CTYPE is_IS.ISO_8859-1
setenv LANG zh_TW.Big5

# 登入時顯示 FreeBSD Tip
[ -x /usr/games/fortune ] && /usr/games/fortune freebsd-tips
```

7.5 csh.logout

這是 Shell csh、tcsh 用的內定 .login 檔案，也就是離開該 Shell 時會載入的設定。在離開 Shell 時，Shell 會載入 /etc/csh.logout 及 ~/.logout 的設定。請 man csh。

7.6 defaults/make.conf

內定的 make.conf 檔案，我們一般不要直接修改 /etc/defaults/make.conf，如果你要設定，請將 /etc/defaults/make.conf 複製一份到 /etc/make.conf ，並修改 /etc/make.conf 來加入你的設定。這樣一來，日後升級時，我們的設定才不會被覆蓋掉。如果 /etc/defaults/make.conf 中的設定和 /etc/make.conf 重覆時，會以 /etc/make.conf 爲主。

如果這個檔案存在的話，會被指令 make 所讀取。它可以讓你在 make 時覆蓋一些設定而不必去修改 source tree。make.conf 檔案一定要使用有效的 Makefile 語法。一般而言，我們會修改一些關於下載檔案的 FTP 站台設定，以從 port 安裝軟體而言，當下達 make install 指令時，make 會去抓一些必要的檔案，如果要去國外抓的話實在太慢了，所以我們會將 FTP 站台設定爲離我最近或最快的站台。讓我們直接看 MASTER_SITE_BACKUP 的部份：

```
# 當使用 port 安裝軟體時，優先使用中央大學資工系及交大資工的 FTP
MASTER_SITE_BACKUP?=    \
        ftp://freebsd.csie.ncu.edu.tw/distfiles/${DIST_SUBDIR}/\
        ftp://freebsd.csie.nctu.edu.tw/pub/distfiles/${DIST_SUBDIR}/
MASTER_SITE_OVERRIDE?=          ${MASTER_SITE_BACKUP}
# 使用的 X window 版本是 4(預設是 3.3.6)
XFREE86_VERSION=4
```

7.7 defaults/rc.conf

這是內定的 rc.conf 檔案，如果要修改，請新增 /etc/rc.conf 來加入你的設定，這樣在日後升級時我們的設定才不會覆蓋。如果 /etc/defaults/rc.conf 中的設定

和 /etc/rc.conf 重覆時，會以 /etc/rc.conf 爲主。另外，當我們執行 /stand/sysinstall 後所做的設定，也會放在 /etc/rc.conf。

7.8 fbtab

大多數的人都是使用虛擬的終端機，我們登入系統時所使用的 tty 是虛擬的終端機 ttyv0、ttyv1 等。FreeBSD 將系統裝置視爲檔案，放在 /dev/，而 console 就是 /dev/console 這個檔，但該檔的擁有者是 root。有的應用程式用要求使用 /dev/console 的存取權限，fbtab 這個檔案是用來定義當你從虛擬的終端機登入時，能自動取得某個裝置的權限。詳細說明請 man fbtab。

7.9 fstab

這個檔案用來定義開機時要掛入的檔案分割區。

# 裝置名稱	掛入點	檔案系統	參數	Dump	Pass#
/dev/ad0s1b	none	swap	sw	0	0
/dev/ad0s1a	/	ufs	rw	1	1
/dev/ad1s1f	/home	ufs	rw	2	2
/dev/ad0s1e	/usr	ufsrw		2	2
/dev/ad1s1e	/var	ufs	rw	2	2
/dev/acd0c	/cdrom	cd9660	ro,noauto0	0	
proc	/proc	procfs	rw	0	0

- 裝置名稱是要掛入的來源，最常用的是 /dev/ 的檔案，我們說過FreeBSD 將裝置視為檔案，所以這裡填的是 /dev/*。裝置也可以是NFS或是其他的虛擬裝置，如 proc,linpro 等。
- 掛入點就是你要將來源掛到什麼地方，其中 swap 沒有掛入點，所以是 none。
- 檔案系統就是要掛入的類型，必須在 kernel 中有定義。一般 FreeBSD 的檔案是 ufs，硬碟要掛入的設定就是 ufs。如果是 cdrom 就是 cd9660。

表10

ufs	本地的 UNIX 檔案系統。
mfs	本地的 memory-based UNIX 檔案系統。
nfs	和 Sun Microsystems 相容的 "Network File System"。
swap	用來作 swapping 的檔案系統。
msdos	DOS 相容的檔案系統。
cd9660	CD-ROM 的檔案系統。
procfs	用來存取執行程序(process)的檔案系統。
kernfs	用來存取核心參數(kernel parameter)的檔案系統。

- 參數依各裝置而有所不同，如果開機時不掛入的話(如 cdrom)，就必須加入參數 noauto。defaults 設定為 rw、dev、exec、auto、nouser、async。可用的參數如下，加上no 則為相反，如 nouser、noauto：

表11

rw	可讀可寫。
ro	只可讀不可寫。
async	所有資料以非同步方式完成。
atime	每次存取動作都更新檔案時間。
auto	能被 mount -a 自動掛入系統。
dev	解譯檔案系統特性與儲存裝置規格。
exec	允許檔案系統中的二進位元檔被執行。
user	允許一般user 掛入。
sync	所有資料以同步方式完成。

sw	swap。
noauto	開機時不掛入。
userquota	使用者磁碟配額限制 (須 kernel 支援 quota)。
groupquota	群組磁碟配額限制 (須 kernel 支援 quota)。

 **dump 表示使用指令 dump 時要備份的檔案系統，0表示不要,1表示要。**

 **pass 這個欄位是給指令 fsck 用的，是檢查的順序。/ 的數字應該是1而其他的檔案系統為2。不需檢查的就是0(如 cdrom,swap 等)。**

7.10 ftpusers

這個檔案用來定義哪些使用者不可以使用 ftp 登入，只要將使用者登入的帳號加入這個檔案中，該使用者就不能使用 ftp 登入系統了。例如，我們要讓使用者 tom 無法使用 ftp 來登入系統，只要在 fupusers 檔案開頭加入該使用者名稱：

```
# $FreeBSD: src/etc/ftpusers,v 1.6 1999/08/27 23:23:41 peter Exp $
#
# list of users disallowed any ftp access.
# read by ftpd(8).
tom
root
toor
daemon
operator
bin
```




```
tty
kmem
games
news
man
bind
uucp
xten
pop
nobody
```

7.11 gettytab

終端機模式設定檔。例如登入前的提示訊息設定等。我們直接來看 default 的部份：

```
# im= 就是別人連上你的機器時會看到的字串，其中 \r\n 表示換行
# 其中的%s %m %h %t 分別對應到 FreeBSD i386 alexwang.com
# ttyp0，
# 如果你不想顯示 FreeBSD ，就把 %s 拿掉。最後一行 if=/etc/issue
# 就是表如果沒有 issue 這個檔的話，就執行 default。
default:\
        :cb:ce:ck:lc:fd#1000:im=\r\n%s/%m (%h) (%t)\r\n\r\n:sp#1200:\
        :if=/etc/issue:
```

7.12 group

設定使用者群組。在 FreeBSD 中，如果使用者要具有 su 成 root 的權限，必須將該使用者加入 wheel 群組中。我們要注意的是每一個群組的編號 (gid) 不可以重覆。請參考第五章「使用者管理」

7.13 host.conf

這個檔案用以設定 DNS 查詢的順序。內定是先查詢 hosts 這個檔，再由 bind 向 DNS Server 查詢。如果有 NIS 的話，還要再加入 nis 那一行。

```
# $FreeBSD: src/etc/host.conf,v 1.6 1999/08/27 23:23:41 peter Exp $
# First try the /etc/hosts file
hosts
# Now try the nameserver next.
bind
# If you have YP/NIS configured, uncomment the next line
# nis
```

7.14 hosts

定義常用機器的 hostname 及 IP，以節省向 DNS Server 查詢的時間。如果你的主機有多個 domain name，將你最喜歡的放在最前面，這樣寄信時才會出現該 domain name。檔案中最好最少要有 localhost 和自己的 domain name。

```
127.0.0.1        localhost.com localhost
192.168.0.1      mydomain.com mydomain
192.168.0.1      mydomain.com.
192.168.0.2      mail.mydomain.com mail
140.115.83.240   bbs.mgt.ncu.edu.tw bbs.mgt
140.115.75.5     bbs.ba.mgt.ncu.edu.tw bbs
```

7.15 hosts.allow

設定允許或拒絕使用本機服務、連線的主機。例如你可以在這裡加入拒絕某台電腦使用 telnet 連線到你的機器。我們將在第十七章「系統安全篇」中會詳細的說明如何使用這個檔案來讓我們的主機更安全。

7.16 hosts.equiv

設定遠端信任主機及使用者，詳細說明請 man hosts.equiv。

7.17 hosts.lpd

可以使用本地列表機的主機，只要加入主機名稱或 IP 即可。

7.18 inetd.conf

設定 inetd 所要提供的服務，要和 /etc/services 搭配設定。如果要開放某一項服務，只要將開頭的 # 符號移除即可。當修改這個檔後，必須重跑 inetd 才會生效。

➔ kill -1 `cat /var/run/inetd.pid`

7.19 localtime

這個檔記載你所在的時區資料，你可以經由 /stand/sysinstall 來設定時區。該程式會自動將 /usr/share/zoneinfo 中合適的檔案複製一份成為 localtime。

7.20 locate.rc

用來設定 local database 的設定。詳細的說明請 man updatedb、man locate。

7.21 login.access

用來設定登入系統使用者的權限，我們可以在這裡設定是否允許使用者從 console 登入、從不同區域登入的權限等。

當某個使用者登入後，會先依本檔中的設定來控制使用者的登入來源。這裡所使用的規則是由上往下比對，先符合者優先，也就是 first match wins，來決定該使用者可以登入或是被拒絕。它的格式是以 ":" 分割成三個欄位：

permission : users : origins

- 第一個欄位應該是 "+" (可以登入) 或 "-" (拒絕登入) 的其中之一。
- 第二個欄位是一個或多個登入的名稱、群組名稱或是 ALL(永遠符合)。
- 第三個欄位是一個或多個 tty 的名稱清單(非網路登入用)、主機名稱、所屬網域(開頭是 "." 的)、主機位址、網際網路編號(結尾是 "." 的)、或是本地的機器 (任何沒有包含 "." 的字串)。如果你有跑 NIS，你可以使用@netgroupname在主機或是使用者的格式中。

我們也可以使用 EXCEPT 的運算符號寫出非常簡潔的規則。群組的設定只有在使用者名稱不符合規則時才會被用到，而且這個程式並不會去查使用者主要的群組是什麼。

7.22 login.conf

這個檔案用來控制不同帳號可以使用的系統資源，它會依照使用者在密碼檔中的 login class 來尋找相對應的 class 設定，如果沒有分類則使用 default 的預設值。修改完該檔後，請記得要執行 cap_mkdb /etc/login.conf 重建系統資料庫。

7.23 mail.rc

用來設定 mail 的參數。當 mail 指令執行時，它會先讀 /usr/share/misc/mail.rc、/usr/local/etc/mail.rc 及 /etc/mail.rc，最後再讀使用者的 ~/mailrc 這個檔。請 man mail，您不必動到這個檔。

7.24 manpath.config

設定 man 指令的參數，如 man 的文件所在等。

7.25 master.passwd

FreeBSD 使用 shadow password 的方式來保護密碼檔，只有 root 才可以讀取編碼後的密碼檔 /etc/master.passwd，而一般使用者只能看到 passwd 檔中的資料。但是這並不是系統用來驗証的檔案，爲了加快速度，FreeBSD 將該檔做成資料庫 /etc/spwd.db 及 /etc/pwd.db， 因此在修改完 master.passwd 後，必須使用指令 pwd_mkdb 來將 master.passwd 做成資料庫檔案。不過一般而言，我會使用 vipw 這個指令來修改master.passwd，vipw 會先將 master.passwd 以預設的文書編輯軟體打開，修改完存檔後，它會視需要自動更新資料庫。

它的格式是：name:password:UID:GID:class:change:expire:fullname:home:shell

- name：使用者帳號名稱，最多可以使用8個字元，不可重覆。
- password：可以是空的，代表不用密碼就可以登入，這樣很危險；也可以是 *，表示不可以登入；上面 vipw 顯示出來的項目中，以使用者 root 而言，他的密碼是使

用 MD5 編碼過的，特徵是開頭為 $1 且看起來比較長；而使用者 tom 的密碼是使用 DES 編碼過的，DES 會將密碼編成一串13個字元的符號。

- UID：使用者代號，每個使用者都不一樣，不可重覆，如果有多個帳號使用同樣的 UID，FreeBSD 會將它當成同一個帳號。編號從 0 到 65535。
- GID：群組代號，編號從 0 到 65535。
- class：除了群組外，class是更有彈性的控制方法，可以針對 /etc/login.conf 中不同的使用者設定來調整每個使用者的可使用的資源設定。
- change：強迫使用者變更密碼的時間，以從1970年到所要變更日期所經過的秒數來表示。你可以使用 date +%s 來求出從1970年到現在時間所經過的秒數，每天為86400秒，以現在時間的秒數加上86400*天數即為你要設定的時間。你可以使用指令expr `date +%s` + 86400 * 30

  來取得30天後的秒數，再將其填入即可。若設為0則表示不使用此功能。
- expire：帳號的有效日期，一樣是以從1970年到到期日所經過的秒數來代表。若設為0則表示不使用此功能。
- fullname：使用者全名，你可以在此鍵入真實姓名。
- home：使用者的家目錄，即使用者登入後的所在目錄。
- shell：使用者的 shell。如果使用 /sbin/nologin 表示該名使用者不可以登入。

7.26 motd

系統登入後，會自動秀出一段文字，稱爲 Message Of The Day(motd)。這一段文字是可以修改的，你可以編輯 /etc/motd 來製作自己的畫面。如

果您不希望 motd 內容出現 FreeBSD 的版本資訊，您可以在 rc.conf 中加入下面內容：

```
update_motd="YES"
```

如此一來，下次您更新 motd 的內容時，系統就不會自動將版本資訊加入 motd 中。

7.27 namedb/

如果你有架 Name Server 的話，你必須設定該目錄下的檔案。詳細說明請參考第十二章DNS伺服器的說明。

7.28 netstart

這個檔案並不會被其他程式所使用，它只是讓你在單人模式下執行，用以啟動網路。而多人模式的網路設定在於 /etc/rc.network。

7.29 networks

用來設定本地網路的資訊。

7.30 newsyslog.conf

這個檔用來定期檢查 /var/log/ 的檔案，設定當 log 到一定大小或是定期的將舊的檔案壓縮備份，並刪除太老舊的檔案。

語法：logfilename [owner:group] mode count size when [ZB] [/pid_file] [sig_num]

表12

logfilename	log 檔名稱
[owner:group]	log 檔擁有人:群組
mode	該 log 檔的權限(檔案屬性)
count	最多計算到多少，例如 cron 是 3 表示將有四個壓縮檔：cron.0.gz, cron.1.gz, cron.2.gz, cron.3.gz。
size	檔案最大到多少即壓縮備份，以 KB 計
when	什麼時候做備份，請 man newsyslog 來看詳細說明。以 @ 為首代表用 ISO 8601 結構的時間格式。以 $ 為首代表使用每天、每週、每月。 一些例子： $D0　每天半夜十二點 $D23　每天 23:00 時 $W0D23 每週日 23:00 $W5D16 每週五 16:00 $MLD0　每月最後一天半夜十二點 $M5D6　每月第五天 6:00
[ZB]	Z 表示要將該檔以 gzip 壓縮起來，B 表示該檔是 binary 檔。
[/pid_file]	pid 檔的絕對路徑，如果有設定，則會送出 sig_num 給該程式。
[sig_num]	要送給該 daemon 程的 signal number，預設是 SIGHUP。

7.31 passwd

FreeBSD 使用 shadow password 的方式來保護密碼檔，只有 root 才可以讀取編碼後的密碼檔 /etc/master.passwd，而一般使用者只能看到 passwd 檔中的資料。但是這並不是系統用來驗証的檔案，爲了加快速度，FreeBSD 將該檔做成資料庫 /etc/spwd.db 及 /etc/pwd.db， 因此在修改完 master.passwd 後，必須使用指令 pwd_mkdb 來將 master.passwd 做成資料庫檔案。不過一般而言，我會使用 vipw 這個指令來修改master.passwd，vipw 會先將 master.passwd 以預設的文書編輯軟體打開，修改完存檔後，它會視需要自動更新資料庫。你不需要改 passwd 的內容，只要改 master.passwd 即可。詳細說明請參考 master.passwd 的說明。

7.32 pccard_ether

用以啓動或停用 PCCARD 網路設備的執行檔。

7.33 periodic/daily

設定每天要執行一次的程式。該目錄下放的是 shell script，如果你想自行增加的話，只需在該目錄下編輯新的 shell script 並將權限設爲可執行即可，或者編輯 /etc/daily.local，加入想要執行的指令。在 /etc/defaults/periodic.conf 中定義了定期執行的設定，以下即爲即爲每個檔案所執行的動作：

表13

檔案	動作	預設執行
100.clean-disks	清理磁碟，要清除的檔案類型可以在 periodic.conf 中設定。	否
110.clean-tmps	清除 /tmp	否
120.clean-preserve	刪除 /var/preserve 中它舊的檔案。	是
130.clean-msgs	清除舊的系統訊息。	是
140.clean-rwho	清除 /var/rwho 中的舊資料。	是
150.clean-hoststat	清除 /var/spool/.hoststat	是
200.backup-passwd>	備份 /etc/master.passwd 及 /etc/group 並比對是否有更動。	是
210.backup-aliases	備份 /etc/mail/aliases	是
220.backup-distfile	備份 /etc/Distfile	是
300.calendar	執行 calendar -a	否
310.accounting	移轉 /var/account/ 的檔案	是
320.distfile	執行 rdist	是
330.news	執行 /etc/news.expire	是
340.uucp	執行 /etc/uuclean.daily	是
400.status-disks	執行 df 及 dump -W	是
410.status-uucp	執行 uustat -a	是
420.status-network0	執行 netstat -i	是
430.status-rwho	執行 uptime	是
440.status-mailq	執行 mailq	是
450.status-security	執行 /etc/security	是
460.status-mail-rejects	統計/var/log/maillog 中記錄拒絕的信件數量。	是
470.status-named	統計 DNS 拒絕記錄。	是
500.queuerun	手動執行 mail queue	是
999.local	執行 /etc/daily.local 中使用者自行定義的指令。	

7.34 periodic/weekly

設定每週要執行一次的程式。該目錄下放的是 shell script，如果你想自行增加的話，只需在該目錄下編輯新的 shell script 並將權限設爲可執行即可。在 /etc/defaults/periodic.conf 中定義了定期執行的設定，以下爲每週會執行的工作：

表14

檔案	動作	預設執行
120.clean_kvmdb	清除過期的 /var/db/kvm_*.db	是
300.uucp	執行 /usr/libexec/uucp/clean.weekly	是
310.locate	執行 /usr/libexec/locate.updatedb	是
320.whatis	執行 /usr/libexec/makewhatis.local	是
330.catman	執行 /usr/libexec/catman.local	否
340.noid	找出沒有擁有人或群組的檔案。	否
400.status_pkg	使用 pkg_version(1) 找出老舊的 package	否
999.local	執行其它在 /etc/weekly.local 中使用者自行定義的指令	

7.35 periodic/monthly

設定每月要執行一次的程式。該目錄下放的是 shell script，如果你想自行增加的話，只需在該目錄下編輯新的 shell script 並將權限設爲可執行即可。在 /etc/defaults/periodic.conf 中定義了定期執行的設定，以下爲每月爲執行的檔案：

表15

檔案	動作	預設執行
200.accounting	執行 ac 指令，統計使用者登入時間	是
999.local	執行其他 /etc/monthly.local 使用者自行定義的指令	

7.36 phones

用來設定遠端主機電話的資料庫，以供指令 tip 使用，詳情請 man tip。

7.37 ppp/

設定 ppp 及 pppd 的設定檔。請參考第六章網路設定中的ADSL 及 modem 的設定。

7.38 printcap

這個檔案定義了列表機的設定，我沒有使用 FreeBSD 列印過，所以請自行參考 man lptcontrol。

7.39 profile

這個檔案是當你使用 bash 爲 shell 時，進入 shell 會讀取的設定。就像 tcsh 所使用的 csh.cshrc 一樣。

7.40 rc

當系統開機時，kernel 會先去載入 /sbin/init，然後 /sbin/init 會去執行 /etc/rc，而 rc 會去執行 /etc/rc.* 的檔案。幾乎所有的設定都在 rc.conf 中，所以你只要去修改 rc.conf 就好了。

7.41 rc.firewall

這個檔案是作防火牆的設定。你可以在該檔中加入你想要的設定，不過前提是你的系統已經準備好防火牆的相關設定了。詳情請參考 13.3 防火牆的設定。

7.42 rc.local

這個檔案是讓你設定開機要執行的程式。你可以在該檔中加入啓動程式的指令，如果 rc.local 不存在，請自行新增。你也可以在 /etc/rc.d 的目錄中，放在你一開機想要執行的 shell script，並將權限改爲可執行。

7.43 rc.*

/etc/ 的其他 rc.* 的檔案，如 rc.i386、rc.network 等，你不必去更動它們。一般而言，你想做的設定都可以在 /etc/rc.conf 中達成。如果你想要知道更多它們的資訊，請 man rc。

7.44 resolv.conf

設定你 DNS 查詢的主機順序。nameserver 的後面加上的 IP 就是 DNS Server 的位址。

7.45 services

這個檔案定義了每個網路服務所使用的 port 及名稱。

7.46 shells

這個檔案定義了使用者可以使用的 shell。只要是可以使用的 shell 都要在該檔案中加入。有些程式為去檢查使用者所使用的 shell 是否在 shells 中，例如 FTP 就不允許非使用 shells 中定義的 shell 的使用者登入。

7.47 syslog.conf

這個檔案定義了系統記錄檔所儲存的位置。

7.48 ttys

定義 tty 的形式及某些 tty 允不允許 root 登入。例入 root 就不能從 ttyp* 登入。有的 tty 後面有加 secure，表示 root 可以從該 tty 登入。

chapter 8 軟體安裝

8.1 概論

傳統上，要在一個 UNIX 系統上安裝其他軟體時，有幾個步驟：

- 下載該軟體，有可能是 binary 檔或是原始碼
- 解壓縮該檔案，通常是以 compress 或 gzip 壓縮的
- 讀一下該目錄中的說明檔，可能是 readme 或是 doc/，來了解如何安裝該軟體
- 如果所下載的是原始碼，可能要先編輯一下 Makefile，接著再編譯該軟體
- 最後再測試與安裝

當然，我們可以在 FreeBSD 上使用傳統的方式來安裝軟體，但是還有更簡單的選擇。FreeBSD提供了 package 和 ports 這二種簡單的安裝軟體方式。

所謂的 pakcage 是別人幫你將程式編譯成 binary 檔，並定義了該安裝在什麼地方。我們只要下載一個壓縮檔，並使用 pkg_add 這個指令就可以快速的將軟體安裝在 FreeBSD 上。這是安裝軟體最簡單的步驟，所安裝的東西也是最標準的，和自己依需求修改並編譯原始碼比較起來較缺乏彈性。

而 port 就是使用原始碼來安裝軟體。我們只要進入 /usr/ports/ 裡想要安裝的軟目錄中，打指令 make install 就可以安裝完成了。FreeBSD 已經幫我們定義了所要安裝讓軟體所須的步驟、所要求的軟體。不管是 package 或是 ports，當安裝的軟體需要依靠其他軟體才能繼續安裝時，它們會自動幫你安裝該軟體。所有安裝好的軟體都將記錄在 /var/db/pkg 中，日後如果我們想要移除軟體時，可以用一個簡單的指令 pkg_delete 加上軟體名稱就可以了。

即然 port 這麼好用，爲什麼 FreeBSD 要同時有 package 和 ports 呢？我們來比較一下 ports 和 package 的優點：

package 的優點：

- 一個已經編譯過的壓縮檔通常比包含原始碼的檔案還要小。
- 使用 pakcage 並不需要再做任何的編譯動作，如果你的電腦速度很慢，在安裝像 KDE、GNOME 等大型軟體時，不用編譯可以省下很多時間。
- 使用 package 來安裝軟體時，你不必事先了解在FreeBSD上編譯時所使用的軟體及其過程。

ports 的優點：

- package 為了要在多數的電腦執行，考慮相容性問題，通常編譯的比較保守。而使用 ports 你可以依自己的系統修改，例如選擇使用 Pentium III 或是 Athlon 的處理器。
- 在編譯 package 時，就已經限制了該軟體的功能，無法再依自己需求擴充。例如 Apache 這套軟體就有許多的功能可以在編譯時掛進來，使用 ports 來安裝時，你
- 可以依自己的需求來加以修改。
- 有些軟體不允許使用 binary 檔的方式散播，只能下載原始碼。
- 有了原始碼，你可以自己修改並加以應用。
- 有的人喜歡擁有原始碼，他們可以讀它、從中學習。

接著我們就針對 package 及 ports 來說明它們的使用方式。

8.2 使用 package

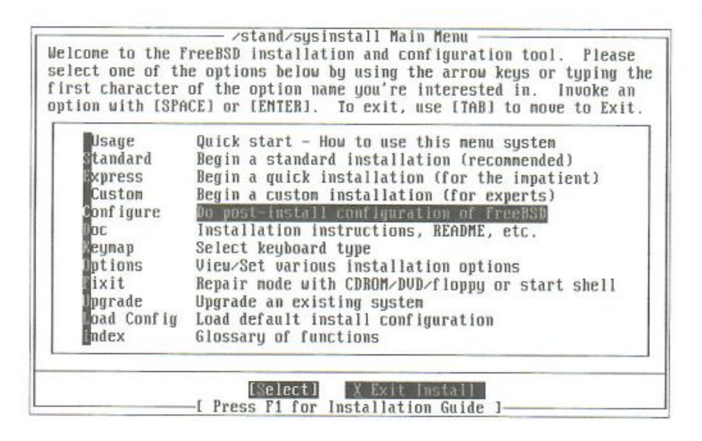
8.2.1 安裝 Package

安裝package 有二種方式，第一種是使用 /stand/sysinstall (也就是我們安裝 FreeBSD 時所看到的畫面) 來安裝，另一種是使用手動安裝。使用 sysinstall 安裝時，我們必需選擇安裝的來源，最常使用的來源是網路及光碟片。而使用手動安裝必須自行抓回所需的檔案，並以指令安裝。以下我們就分別針對這二種方式來說明：

方式一：使用 /stand/sysinstall

假設我們要安裝在 FreeBSD 上收信的軟體 pine，首先我們要執行 sysinstall 以進入安裝時的畫面：

```
# /stand/sysinstall
```


/stand/sysinstall Main Menu
Welcome to the FreeBSD installation and configuration tool. Please select one of the options below by using the arrow keys or typing the first character of the option name you're interested in. Invoke an option with [SPACE] or [ENTER]. To exit, use [TAB] to move to Exit.

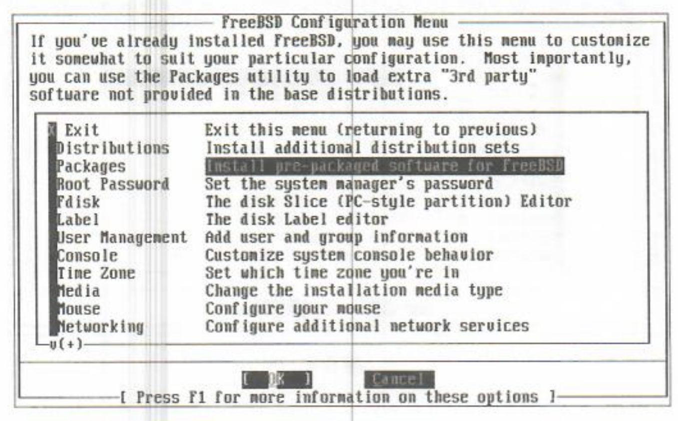
Usage	Quick start - How to use this menu system
Standard	Begin a standard installation (recommended)
Express	Begin a quick installation (for the impatient)
Custom	Begin a custom installation (for experts)
Configure	Do post-install configuration of FreeBSD
Doc	Installation instructions, README, etc.
Keymap	Select keyboard type
Options	View/Set various installation options
Fixit	Repair mode with CDROM/DVD/floppy or start shell
Upgrade	Upgrade an existing system
Load Config	Load default install configuration
Index	Glossary of functions

[Select] X Exit Install
[Press F1 for Installation Guide]

圖8-1



接著選取 Configure選項，進入圖 8-2 的畫面：



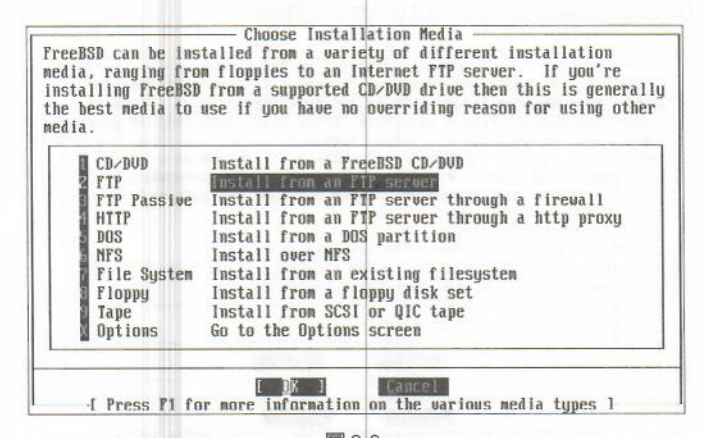
```
FreeBSD Configuration Menu
If you've already installed FreeBSD, you may use this menu to customize
it somewhat to suit your particular configuration.  Most importantly,
you can use the Packages utility to load extra "3rd party"
software not provided in the base distributions.

X Exit             Exit this menu (returning to previous)
Distributions      Install additional distribution sets
Packages           Install pre-packaged software for FreeBSD
Root Password      Set the system manager's password
Fdisk              The disk Slice (PC-style partition) Editor
Label              The disk Label editor
User Management    Add user and group information
Console            Customize system console behavior
Time Zone          Set which time zone you're in
Media              Change the installation media type
Mouse              Configure your mouse
Networking         Configure additional network services
v(+)

[ OK ]      Cancel
[ Press F1 for more information on these options ]
```

圖 8-2

我們選 Packages 選項來安裝 package：



```
Choose Installation Media
FreeBSD can be installed from a variety of different installation
media, ranging from floppies to an Internet FTP server.  If you're
installing FreeBSD from a supported CD/DVD drive then this is generally
the best media to use if you have no overriding reason for using other
media.

1 CD/DVD         Install from a FreeBSD CD/DVD
2 FTP            Install from an FTP server
3 FTP Passive    Install from an FTP server through a firewall
4 HTTP           Install from an FTP server through a http proxy
5 DOS            Install from a DOS partition
6 NFS            Install over NFS
7 File System    Install from an existing filesystem
8 Floppy         Install from a floppy disk set
9 Tape           Install from SCSI or QIC tape
X Options        Go to the Options screen

[ OK ]      Cancel
[ Press F1 for more information on the various media types ]
```

圖 8-3

在圖8-3 中，我們必須選擇安裝來源，在這裡我們選擇 FTP ，從 FTP 中我們可以找到較多的軟體。選擇了 FTP 之後，將出現圖8-4的畫面，讓我們選擇要使用哪一個 FTP 站台：

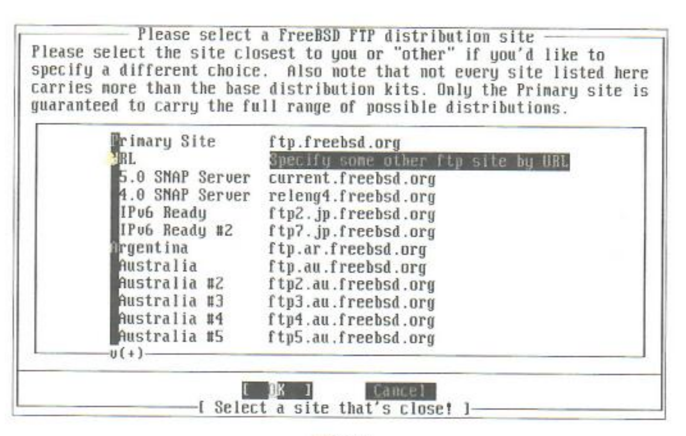


圖 8-4

我們選擇「URL」來自訂要使用的 FTP 站台。接著便會出現一個要求我們輸入站台位址的視窗，如圖8-5所示，假設我們要使用交大資工的站台，則輸入 freebsd.csie.nctu.edu.tw/pub/i386/：

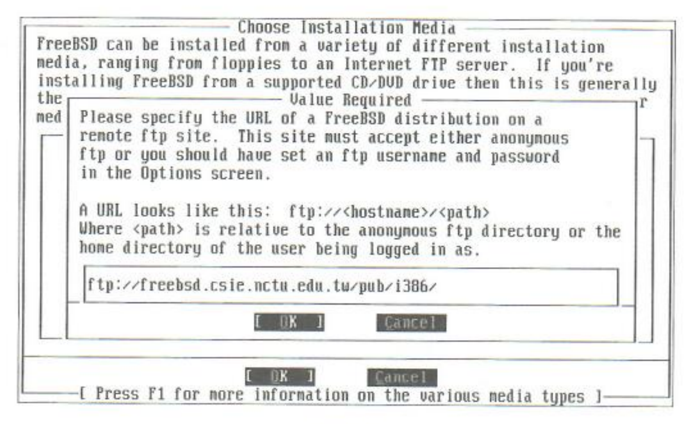


圖 8-5

輸入站台後，會問您是否要使用目前的網路設定。如果我們已經連上網路，則選 YES，否則請選 NO 來設定網路。接著會出現一個軟體分類選單，這一份分類選單將各個軟體分門別類放在不同的選項下，其中 All 是所有軟體的所在，如圖8-6。

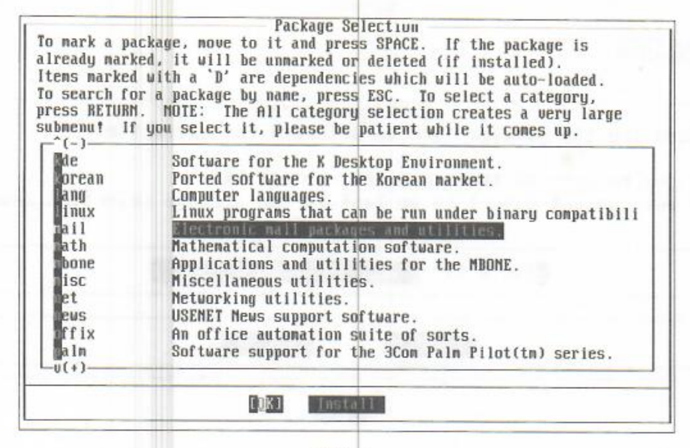


圖8-6

我們以安裝郵件軟體 pine 為例，由於 pine 位於 mail 分類下，所以我們選擇mail。如果您要安裝中文版本的 pine ，應該選擇 Chinese 選項而非 mail。選擇了 mail 之後，將出現 mail 分類下的所有軟體，我們選 pine-4.44 這一項，如圖8-7：

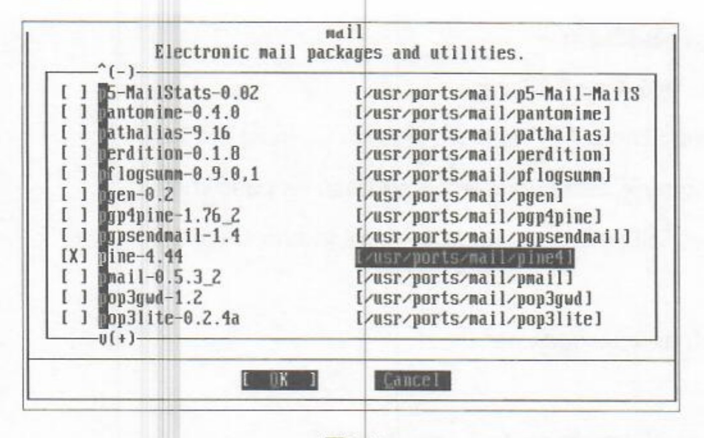


圖8-7

選了pine 之後，就可以選 OK 回到前一個分類畫面，接著按照這種方式選了其他我們要安裝的軟體之後，就可以選「Install」來安裝了。選了 Install 之後，將出現所有我們已選取的軟體列表，如圖8-8，如果要繼續

安裝則選 OK 即可。

```
                        Package Targets
 These are the packages you've selected for extraction.

 If you're sure of these choices, select OK.
 If not, select Cancel to go back to the package selection menu.

              pine-4.44   [/usr/ports/mail/pine4]

                  [  OK  ]        Cancel
```

圖 8-8

方式二：使用手動安裝

如果以手動的方式安裝，我們必須先取回所要安裝的package。只要是 package ，它的副檔名就是 .tgz。我們可以用 pkg_add 這個指令來安裝它。下面是一個簡單的範例，使用 package 來安裝 lsof-4.45.4.tgz：

```
# ftp -a ftp.FreeBSD.org
Connected to ftp2.FreeBSD.org.
220 ftp2.FreeBSD.org FTP server (Version 6.00LS) ready.
331 Guest login ok, send your email address as password.
230- The FreeBSD mirror at Tele Danmark Internet.
230-
230- Contact: beastie@tdk.net
230-
230- Use wisely.Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /pub/FreeBSD/ports/packages/sysutils/
250 CWD command successful.
ftp> get lsof-4.56.4.tgz
```

```
local: lsof-4.56.4.tgz remote: lsof-4.56.4.tgz
200 PORT command successful.
150 Opening BINARY mode data connection for 'lsof-4.56.4.tgz' (92375 bytes).
100% |**************************************************| 92375 00:00 ETA
226 Transfer complete.
92375 bytes received in 5.60 seconds (16.11 KB/s)
ftp> exit
# pkg_add lsof-4.56.4.tgz
```

要使用 package 安裝軟體，首先必須取得想要安裝的軟體。我們可以先 ftp 到各大學 FTP 站台去取得。packaeg 的副檔案是 .tgz，可以在各 FTP 站台的 i386/packages 中取得。以交大資工的FTP站而言是放在 ftp://freebsd.csie.nctu.edu.tw/pub/i386/packages ；而中央資工的 FTP 是放在 ftp://freebsd.csie.ncu.edu.tw/i386/packages 。當進入該目錄後，我們會發現還有一堆目錄，您可以依您的系統版本選擇要使用哪一個目錄。例如系統是 4.5-Release 或是 4.5-STABLE，我們就可以選擇進入 packages-4.5-release 這個目錄。進入這個目錄後，又有一堆目錄，這裡的目錄結構和你系統中 /usr/ports/ 下的目錄一樣，每個目錄都是軟體的分類，而 All 這個目錄是所有軟體。

如果您只知道想要安裝的軟體名稱，卻不知道版本及完整的檔名，例如您要下載 popa3d 這個軟體，但不知道是哪一版的，你可以先進入 All 的目錄下，再以下列方式查詢：

```
ftp> ls popa3d*
227 Entering Passive Mode (140,113,209,200,159,54)
150 Opening ASCII mode data connection for /bin/ls.
-r--r--r-- 1 FTP CSIE 22259 Sep 16 07:31 popa3d-0.4.tgz
```

```
226 Transfer complete.
ftp> get popa3d-0.4.tgz
```

找到了想要下載的版本是 0.4，接著就以 get 指令去取回該軟體，最後下 exit 離開。

接著你就可以使用 pkg_add popa3d-0.4.tgz 來安裝該軟體。

8.2.2 管理 Package

如果我們後悔了，想要移除該軟體，可以下指令 pkg_delete popa3d-0.4 來移除 popa3d-0.4 這套軟體，所有我們安裝過的軟體都會出現在 /var/db/pkg 的目錄中。

我們可以使用 pkg_info 這個指令來得到軟體的資訊。例如在我們下載完一個 package 後，你想要知道這個軟體的資訊，以 popa3d-0.4.tgz 而言，如果我們想知道它的資訊，你可以下：

```
# pkg_info popa3d-0.4.tgz
```

您也可以只打 pkg_info 來得知所有你安裝過的軟體有哪些。



8.3 使用 ports

如果你要使用 ports 安裝軟體，你必須先確認 /usr/ports 這個目錄是否有安裝。如果沒有的話，使用 /stand/sysinstall 來安裝 ports 的目錄：

1. 以 root 執行 /stand/sysinstall
2. 選擇 Configure 後按 Enter
3. 選擇 Distributions 後按 Enter
4. 選擇 ports 後按空白鍵
5. 選擇 Exit 後按 Enter
6. 選擇你要從 CDROM 或 FTP 安裝等
7. 跟著選單照做，最後離開 sysinstall

或者我們也可以到 http://www.freebsd.org/ports/ 去手動抓回 port.tar.gz 這個檔案，將它放在 /usr/ 下。並以下列指令來安裝：

```
cd /usr
tar zxvf port.tar.gz
```

現在可以進入 /usr/ports 的目錄中，安裝軟體了。

通常每一個軟體都有一個獨立的目錄，而目錄中都存在著一些檔案，每個檔案都有其特定用途，我們簡列如下：

表16

Makefile	安裝軟體的編譯設定，您可以修改這個檔案來設定我們在編譯即安裝軟體時的參數。
distinfo	說明安裝所需要的檔案及其 MD5 的檢查資料。
pkg-comment	簡單的軟體描述。
pkg-descr	較詳細的描述，我們通常可以在裡面找到該軟體網頁的位置，使我們能到該網頁得到更多資訊。
pkg-plist	列出軟體將安裝的清單，安裝後會放在硬碟中的什麼地方。

如果您想安裝某一個軟體，卻不知道它的目錄位置，您可以使用 whereis 這個指令來找出它來。例如我們想安裝 qpopper ，可以使用 whereis qpopper 來找出它所在的目錄。 或者果我們只知道某個程式的關鍵字，確不知道它放在哪個目錄，我們可以使用下列指令：

```
cd /usr/ports
make search key='關鍵字'
```

進入該目錄後，最簡單的安裝方式是直接打 make install，系統就會自動去網路上抓取需要的軟體回來安裝。安裝 ports 時，make 時找檔案的順序是：先去 /usr/ports/distfiles 、再去找/cdrom/ports/distfiles、最後是網路中下載。如果您不使用網路安裝的話，您可以自己去抓回軟體，並將它放在 /usr/ports/distfiles/ 下，這樣子在我們打 make install 時，就不會去網路上抓取檔案。如果您所需檔案存在光碟中，在安裝軟體之前，必須先將光碟機 mount 在 /cdrom 中。例如，本書的使用者只需將所附的第二片光碟放入光碟機，並使用指令 mount /cdrom 就可以安裝本書所提及的軟體了。或者，您也可以將光碟中 /ports/distfiles 目錄內容複製到 /usr/ports/distfiles 目錄中。

當使用網路取得檔案時，預設抓取檔案的伺服器通常在國外，因此，您可以修改 /etc/make.conf 來指定使用國內的 FTP 站台，例如編輯 /etc/make.conf 並加入：

```
MASTER_SITE_BACKUP?= \
        ftp://freebsd.csie.ncu.edu.tw/distfiles/${DIST_SUBDIR}/ \
        ftp://freebsd.csie.nctu.edu.tw/pub/distfiles/${DIST_SUBDIR}/
MASTER_SITE_OVERRIDE?=   ${MASTER_SITE_BACKUP}
```

當安裝完 ports 後，我們可以再下指令 make clean 來清除編譯過程產生的檔案，建議最好這麼做，否則有的過程中產生大量檔案可是很驚人的。如果您安裝了一堆軟體之後，才想到之前沒有 make clean，沒關係，在安裝 ports 時，編譯過程的檔案都存在於該軟體目錄下的 work 目錄中。我們可以使用下列指令來找出所有未 make clean 的軟體，並將暫存資料刪除：

```
# find /usr/ports -depth -name work -exec rm -rf {} \;
```

如果您使用網路安裝，它會將所下載的原始碼存在 /usr/ports/distfiles 中，當你下 make clean 後，並不會將它們清除。

當你安裝完後，想要移除該軟體時，只要在該軟體的 ports 目錄中打 make deinstall 即可。請注意，不要在 /usr/ports 的目錄中打 make deinstall，這樣可是會將 "所有" 軟體都移除喔。

還有一些較不常用的 make 方式，簡述如下：

make fetch	抓回所需的原始檔。
make fetch-list	顯示安裝所需的檔案。
make checksum	抓回原始檔並以 MD5 檢查其正確性。
make extract	抓回並解開原始檔。
make configure	進行組態，但不繼續編譯。
make all install	抓回原始檔、編譯且安裝。
make reinstall	若先前發生意外中斷，以此命令繼續嘗試安裝。
make package	將做好的 ports 打包製作成 packages。

如果安裝完新的軟體之後，如果使用的 Shell 是 Csh 或 Tcsh，我們可能必須執行指令 rehash 來重建 hash table，之後才能在所設定的指令路徑中

找到剛安裝的程式，不然的話就必須輸入該程式的完整路徑或重新登入才能使用。

我們可以在 /var/db/pkg 的目錄中看到我們已安裝的軟體，每一個軟體有一個目錄，目錄中存放著軟體安裝的資訊，包含了軟體說明、安裝到哪些目錄中。有的軟體要安裝前，會要求先安裝某一套軟體，如果你事先沒有安裝它所要求的軟體，通常該軟體會自動幫你安裝。所以我們會在 /var/db/pkg 下看到一些不是我們主動安裝的軟體。既然軟體之間可能會相互依賴，我們要如何得知這些軟體彼此間的關係呢？pkg_tree 這套軟體可以讓我們檢視軟體間的關係。我們可以使用 port 來安裝這套軟體：

```
cd /usr/ports/sysutils/pkg_tree
make install clean
```

之後我們就可以使用 pkg_tree | more 來看各個軟體之間的關係了 (別忘了要 rehash 喔)。

chapter 9 X Window 的使用

9.1 安裝 X Window

X Windows 是在 UNIX 系統下的視窗軟體，目前的版本 4.1.0。另外我們會再加裝視窗管理軟體，如果沒有了它，X Windows 就只能看到白白一片。在眾多的視窗管理軟體中，我們選用 KDE，因為它提供了很多常用的工具，例如瀏覽器、Office 軟體等。因此，我們將安裝 X Windows + KDE 2 及並將其介面中文化。

Step1：安裝X Window

首先安裝 X Window，請先編輯 /etc/make.conf，加入下面這一行：

```
XFREE86_VERSION= 4
```

接著使用 ports 來安裝 x Window：

```
# cd /usr/ports/x11/XFree86-4
# make install clean
```

安裝的時間需要很久，以CPU 233 Hz 的速度要五個小時左右。

Step2：安裝KDE 2

我們同樣使用 ports 來安裝 KDE 2：

```
# cd /usr/ports/x11/kde2
# make isntall clean
```

安裝 KDE 2 需要更長的時間，CPU 速度慢一點的大概要花了十幾二十個鐘頭吧。當然，所需的硬碟空間也不小，筆者在安裝完 KDE 2 之後，

尚未 make clean 時要 2.2 GB，安裝過程中有將近 1 GB 的暫存檔，所以一定要 make clean。 安裝到一半時會問您要支援哪些列表機，您可以加入你的列表機。

Step3：安裝中文化字型

接著我們要將 KDE 中文化，到這裡速度會快很多。如果要在 X Window 中看到中文的選單，必須先裝中文字型 kcfont (國喬字型) 及 arphicttf (文鼎字型)，及中文訊息檔 (i18n)：

```
# cd /usr/ports/chinese/kcfonts
# make install clean
# cd /usr/ports/chinese/arphicttf
# make install clean
# cd /usr/ports/chinese/kde2-i18n
# make install clean
```

Step4：產生設定檔

我們接著要執行 XFree86 -configure 來產生設定檔 XF86Config.new，然後將它搬到 /etc/X11/：

```
# XFree86 -configure
# mv ~/XF86Config.new /etc/X11/XF86Config
```

接著編輯 /etc/X11/XF86Config ，在 Section "Module" 中加入 load "xtt"。並在 FontPath 區段最前面加入下面二行，以期使 X Window 能找到正確的字型路徑：

```
FontPath  "/usr/X11R6/lib/X11/fonts/TrueType"
FontPath  "/usr/X11R6/lib/X11/fonts/local"
```

再來設定螢幕的解析度等，先設定一下螢幕，找到 Section "Monitor" 的部份：

```
Section "Monitor"
  Identifier "Monitor0"
  VendorName "Monitor Vendor"
  ModelName "Monitor Model"
  Horizsync 30-70
  VertRefresh 50-120
```

我們在這個區段加入了最後二行關於螢幕水平及垂直更新頻率。接著要設定螢幕的解析度，我希望以 16bit 色彩顯示，800x600 而且不要虛擬桌面，找到 Section "Screen" 的部份：

```
Section "Screen"
  ......略...
  DefaultColorDepth 16
  .......略......
  SubSection "Display"
      Depth 16
      Modes "800x600" "1024x768"
      Virtual 800 600
      ViewPort 0 0
  EndSubSection
```

我們加入了 DefaultColorDepth 16，表示內定以16 bit的色彩顯示，並找到 Depth 16 的部份，加入了 Modes、Virtual、及 ViewPort。這三行表示可以用

800x600 或 1024x768 的解析度、虛擬桌面為 800x600。接著存檔離開。

Step5：設定字型

再來新增 ~/.fontguess，內容如下，以使 Qt lib 在替代字型時不會發生錯誤：

```
[big5-0][gb2312.1980-0][ksc5601.1987-0]
Helvetica     ming      ming        gulim
times                   ming        ming        batang
courier                 ming        ming        dotum
utopia                  ming        ming        gulim
clean                   ming        ming        gulim
ming                    helvetica   helvetica   helvetica
kai                     helvetica   helvetica   helvetica
```

編輯 ~/.xftconfig，加入：

```
dir "/usr/X11R6/lib/X11/fonts/TrueType"
# Danny:
# set the AA for different fonts
#
# most TT fonts do not need to be aliased between
# 8 and 15 points, although this might be a matter of taste.
match
    any size > 8
    any size < 15
edit
    antialias = false;
```

Step 6：進入 x Window

爲了一進入 x Window 即有 KDE 要先編輯 ~/.xinitrc 加入下列六行：

```
#!/bin/sh -
export LC_ALL=zh_TW.Big5
export LC_CTYPE=zh_TW.Big5
export LANG=zh_TW.Big5  # 設定使用中文
export QT_XFT=true  # For Anti-Alias function
exec /usr/local/bin/startkde
```

接著執行 rehash;startx 進入 x Window，進去後看到字都是 ??? 沒關係，在下方工具列中有一個登出的按鈕(在鎖頭的下方小小的圓形) 來登出，或者是同時按 Ctrl+Alt+backspace鍵 (倒退鍵) 來離開。然後編輯 ~/.kde/share/config/kdeglobals，找到 [Locale] 的段落改成如下，如果沒有 [Locale] 區段則在文件最下方自行增加：

```
[Locale]
Charset=big5-0
Country=tw
Language=zh_TW.Big5
```

存檔離開後就可以再一次以 startx 來進入 X Windows。現在應該可以看到完整的 KDE 了。

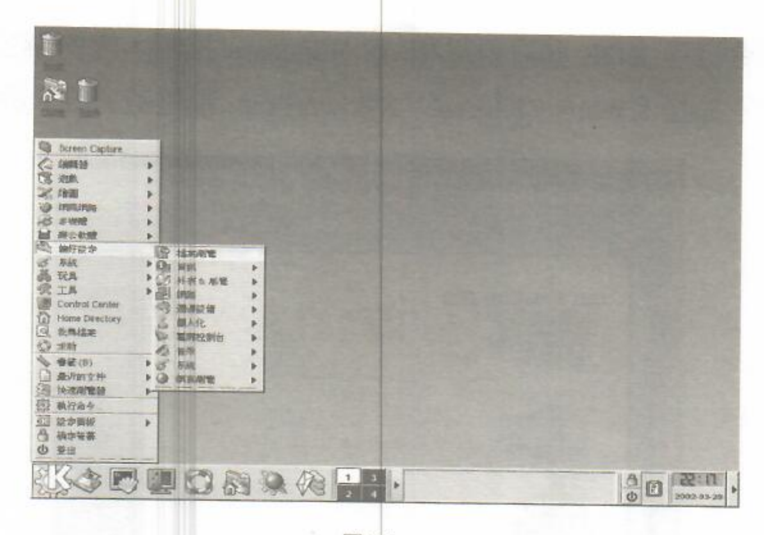


圖 9-1

如果您在X Window中無法使用滑鼠，請執行 /stand/sysinstall來設定滑鼠，設定的位置在 [Configure]->[Mouse]，先設定 [Type] 選擇滑鼠的類型，再選 [Enable] 讓一開機即驅動滑鼠。在 KDE2 中有許多的附屬軟體，從簡單的文字編輯器、繪圖軟體，到常用的辦公室軟體、瀏覽器及郵件軟體都有。由於是圖形介面，您可以自行摸索嘗試。KDE 的瀏覽器是 Konqueror，它的使用介面和 IE 差不多，除了是網頁瀏覽器外，也結合了檔案總管的功能，圖9-2 即 Konqueror的畫面：

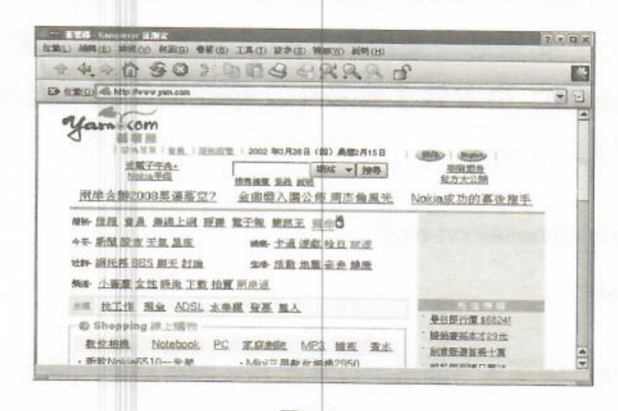


圖 9-2

您也許會發現 KDE 簡直可以和MS Windows 抗衡，它的辦公室軟體功能齊全，不論是 KWord、KExcel、KPowerPoint 都是威力強大的軟體。



圖 9-3

9.2 X Window下的中文軟體

9.2.1 中文終端機

KDE 所附的終端機 Konsole (位於桌面下方有貝殼的黑色螢幕圖示)並不支援中文的顯示，如果您要使用中文的終端機來連上 BBS 站，必須安裝其他軟體。FreeBSD 中收錄了許多中文終端機軟體，例如 Eterm、crxvt 等，這裡我們將安裝 crxvt：

```
# cd /usr/ports/chinese/rxvt-big5
# make install
```

接著請在 ~/.cshrc 中加入下面這一行來使 rxvt 能看到中文檔名：

```
alias  rxvt     'env LC_CTYPE=en_US.ISO_8859-1 rxvt'
```

安裝完畢啓動 X 視窗後，我們可在 [開始選單]->[執行命令] 中，輸入 rxvt 來啓動中文的終端機。如圖9-4所示：

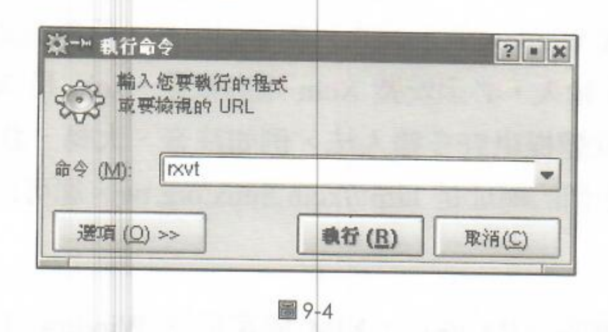


圖 9-4

啓動 rxvt 後，就可以使用中文的終端機畫面了：



圖 9-5

9.2.2 中文輸入

我們安裝的 X Window 目前爲止只能看到中文，但無法使用中文輸入，如果要使用中文輸入，必須安裝 Xcin 這套軟體。xcin 是 X Chinese Input 的縮寫，這個軟體提供許多輸入法，例如注音、大易、倉頡、簡易、酷音、行列等。他們的網址是 http://xcin.linux.org.tw，您可以在這裡獲得更多資訊。

xcin 採用標準的 XIM 協定，XIM 協定是 X Window 下中文輸入的標準，只要支援 XIM 的軟體，我們都可以使用 xcin 來輸入中文。而在 KDE 中，除了 Konsole 外，其他常用的軟體都支援 XIM中文輸入。

安裝 xcin十分容易，我們可以使用 ports 來安裝：

```
# cd /usr/ports/chinese/xcin25
# make install
```

安裝完 Xcin 後，我們還要修改 ~/.xinitrc 來加入中文輸入法的設定，請使用文書編輯軟體來編輯 ~/.xinitrc 這個檔案：

```
#!/bin/sh -
export LC_ALL = zh_TW.Big5
export LC_CTYPE = zh_TW.Big5
export LANG = zh_TW.Big5
export QT_XFT = true
# 加入下列二行
export XMODIFIERS = @im = xcin
xcin2.5&

#啓動 KDE
exec /usr/local/bin/startkde
```



修改完後存檔，接著進入 X Window 您將看到輸入法的視窗，如圖 9-6：

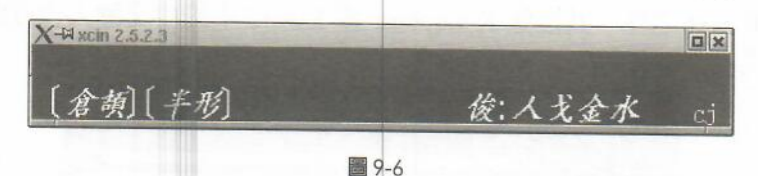


圖 9-6

然後我們就可以執行支援其他軟體來輸入中文了。例如我們開啓 rxvt 的視窗後，就可以使用 Ctrl+Space 來切換中英文輸入法。如果您發現開啓 rxvt 時，無法切換輸入法，請在啓動 rxvt 時加上參數 rxvt -im xcin 即可：

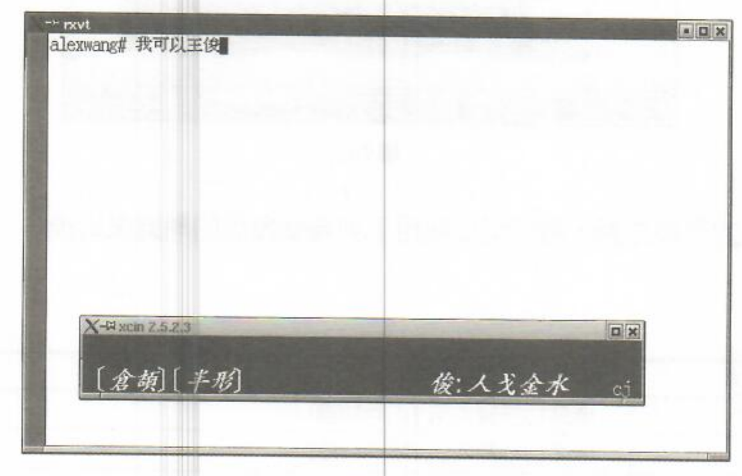


圖 9-7

除了使用 rxvt 之外，其他像 Konqueror 或 Kword 下的中文輸入也沒問題，如圖 9-8。

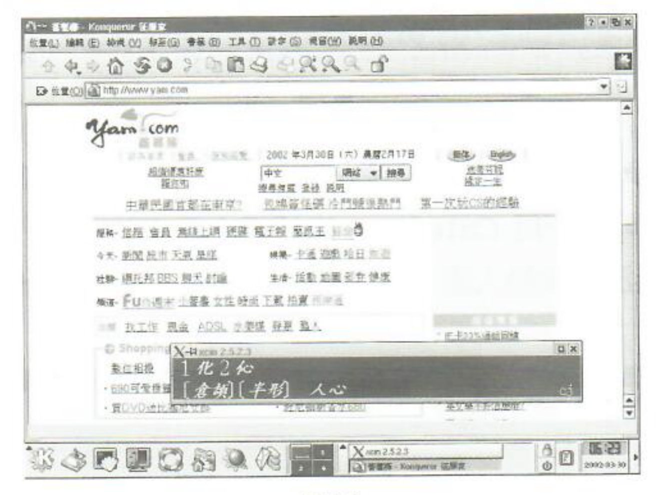


圖 9-8

在輸入法的切換方面，我們可以使用下列幾個預設的熱鍵來切換：

表17

Ctrl+Space	中文 / 英文的切換
Ctrl+Shift	依序切換輸入法 (正向切換)。
Shift+Ctrl	依序切換輸入法 (反向切換)。
Ctrl+Alt+數字	選擇輸入法，數字部份由 1~8

如果您覺得輸入法的視窗太大很佔空間，我們可以修改 xcin 的設定檔，將視窗縮小。設定檔的位置是 /usr/X11R6/etc/xcinrc，使用文書編輯軟體打開後，找到 "OVERSPOT_WINDOW_ONLY" 的部份，並修改如下：

```
(define OVERSPOT_WINDOW_ONLY   "YES")
```



接著重新啟動 X 視窗，當要輸入中文時，只要以 Ctrl+Space 就可以打開如圖9-8 的畫面：

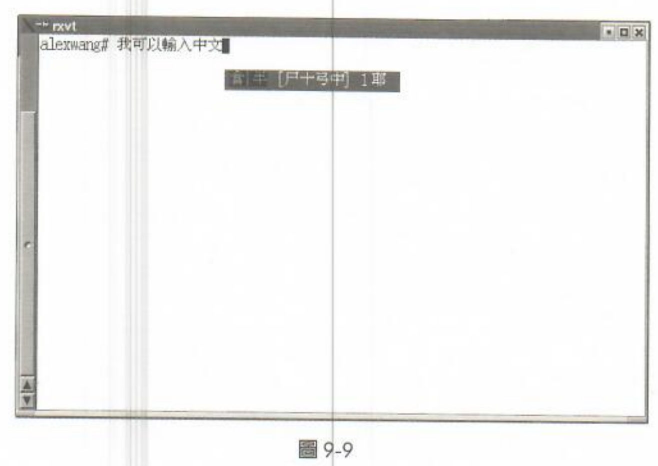


圖 9-9

不過，在 xcin 中似乎有個小 bug，就是使用了 xcin 後，要登出 KDE 時，可能會發現無法登出，最後只能以 Ctrl+Alt+Backspace 來離開 X Window。如果您也有這樣的問題，請修改 /usr/X11R6/etc/xcinrc 將 Root OverTheSpot 的功能拿掉，也就是說我們不能使用小型的輸入法視窗了。我們使用文書編輯軟體打開 /usrX11R6/etc/xcinrc 後，找到下面的內容：

```
; XIM Input Style Adjustments.
; 在下面這一行前面加上一個符號 "#"
#(define INPUT_STYLE          '(Root OverTheSpot))
(define OVERSPOT_USE_USRCOLOR   "YES")
(define OVERSPOT_USE_USRFONTSET "NO")
(define OVERSPOT_WINDOW_ONLY    "NO")
```

如果您發現只有在 rxvt 中可以輸入中文，而在其他的 KDE 軟體下都無法切換到其他的輸入法，很有可能是您LC_CTYPE的設定有問題，請參考前一節中 ~/.xinitrc 的設定內容，將 LC_CTYPE 設為 zh_TW.Big5。

chapter 10

網頁伺服器

apache 是 UNIX 系統中普遍使用的網頁伺服器軟體。目前網際網路中，有超過百分之五十的伺服器是使用 apache 來提供網頁瀏覽的服務。這裡我們將介紹如何安裝一個功能完整的網頁伺服器。

如果你的網頁伺服器只要用來放純粹的 HTML 檔，不要執行其他的功能，如 PHP、MySQL、SSL等，你只需到 /usr/ports/www/apache13 的目錄中，執行 make install 即可迅速的安裝 apache。但是這樣的伺服器太過於陽春了，使用 apache 自然要使用 PHP 才有意思。PHP 是一個用來寫網頁程式的軟體，就像 ASP、JAVA servlet、CGI 等等有類似的用途。不同的是 PHP 十分容易學習，程式碼也很簡潔，速度更是沒話說。如果你有些微的程式語言基礎，不出二個禮拜，你就能對 PHP 有十足的認識，並且可以自己寫出留言版、權限控制等簡單的程式。

如果要使用 PHP，那你一定也要使用 MySQL。MySQL 是一套資料庫系統，它的功能及速度都令人讚賞。apache+PHP+MySQL 是近年十分流行的組合，使用 PHP+MySQL，你可以製作出網頁的各式資料庫，如會員管理、產品資料庫等等。總之，我十分建議使用 apache+PHP+MySQL 的組合，就算目前不會用到，不久的將來也會使用它們的功能。全部一股腦的裝起來，省得日後麻煩。在安裝 PHP 時，我也建議你順便安裝 GD 等軟體來付予 PHP 繪圖的能力，例如從資料庫中取出資料來繪製統計圖表等。

另外，如果你要在網頁上執行 CGI 的話，我們會介紹 suEXEC 的設定。傳統上，當使用者執行 CGI 時，系統會以網頁伺服器執行者的身份來執行 CGI。內定的使用者是 nobody，這樣的執行方式有一些缺點。因為所有的 CGI 程式都必須要設定為可以執行，但是如果是以 nobody 的身份執行的話，該 CGI 程式就必須要讓 nobody 有執行的權限。UNIX 的權限控制有三個等級，分為檔案的擁有者、和權有者同一群組的人、其他

人，而網頁伺服器的執行者通常不是檔案的擁有者，如果系統中有其他的使用者也要執行 CGI 程式，他們都必須把 CGI 程式的權限開放給所有人，這樣子在系統中的所有人都可以執行該程式。更甚者，如果 CGI 程式有要求讀寫檔案的話(例如留言版程式)，那麼被讀寫的檔案也必須讓所有使用者都可以讀寫，也就是說其他人都可以刪除別人的檔案。因此，我們利用 suEXEC 來讓 CGI 程式在執行時是以檔案擁有人的身份執行，也就是說 CGI 程式的權限設定只要設爲擁有者可以讀、寫及執行，不必開放給其他的人使用。總而言之，如果你的網頁伺服器有必要執行 CGI 的話，最好安裝 suEXEC。

最後，我們也會加入 SSL 連線。一般的 http 要求都是以明碼傳送資料，資料傳送的過程中很容易被竊聽。如果你有一些需要輸入密碼的網頁，建議改用 https 連線，也就是用 SSL 連線的方式，將資料重新編碼加密，來增加安全性。

這裡我們將介紹如何安裝 apache，同時使 apache 擁有 php、MySQL、ssl、suEXEC 的功能。如果你不需要某一項功能，你只需要跳過該項設定即可。

10.1 安裝 MySQL

我們可以使用 ports 來安裝 MySQL，但爲了使用最新版的 MySQL，所我們自行抓回原始檔來編譯。先到 http://www.mysql.com 取得最新版的 MySQL Source Package，本文撰寫時的最新版本是 mysql-3.23.46.tar.gz，讀者可以在第二片光碟的 /ports/distfiles 目錄中找到該檔案。你也可以在國內的 FTP 站台去取得，例如 ftp://freebsd.csie.ncu.edu.tw/distfiles/ 取得

mysql-3.23.46.tar.gz。檔案名稱數字的部份就是版本的名稱，數字越大表示版本越新。

先新增一個使用者 mysql 以供 MySQL 使用，編輯 /etc/group 加入下列一行：

```
mysql:*:100:
```

執行 vipw，加入下列一行：

```
mysql:*:100:100::0:0:Mysql user:/usr/local/mysql:/sbin/nologin
```

取回檔案後，執行下列指令以解壓縮，並安裝，下列指令中最後有 \ 表示下一行接續該行：

```
# tar zxvf mysql-3.23.49.tar.gz
# cd mysql-3.23.49
# ./configure --prefix=/usr/local/mysql --with-low-memory \
  --with-named-thread-libs=-lc_r --with-charset=big5
# make
# make install
# scripts/mysql_install_db
```

這樣就完成了 MySQL 的安裝了。上面指令中的 ./configure 的參數，prefix 表示要安裝的目錄，建議安裝在 /usr/local/mysql 中。接著更改 mysql 目錄的擁有者：

```
# chown -R mysql.mysql /usr/local/mysql
```

接下來就試驗一下可不可以執行。接著啓動 MySQL 並設定密碼：

```
# cd /usr/local/mysql/share/mysql
# ./mysql.server start
# cd /usr/local/mysql/bin
# ./mysql mysql
```

若安裝成功，你將看到以下畫面：

```
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14 to server version: 3.23.49

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql>
```

MySQL 剛安裝完成時，並未設定 root 的密碼，因此我們接著要設定 root 的密碼並即時更新設定：

```
mysql> UPDATE user SET password=password('密碼')
    where user='root';
Query OK, 0 rows affected (0.00 sec)
Rows matched: 2  Changed: 0  Warnings: 0

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.02 sec)
```

如果您有其他使用者要加入也可以加入，最好不要讓使有人都有對所有資料庫有全部的權限。例如，我的網頁資料庫名稱是 www，而管理者是

jack，我只要讓它對 www 這個資料庫有某部份的權限且密碼是 mypwd，可以使用下列的設定：

```
mysql> GRANT SELECT,INSERT,UPDATE,DROP,CREATE,DELETE,INDEX
    ON www.* TO jack@localhost IDENTIFIED BY 'mypwd';
mysql> FLUSH PRIVILEGES;
```

以上指令及 MySQL 更詳細的設定說明，我們會在第十五章資料庫系統中加以說明。最後請以 exit; 來離開 MySQL。

開機時要自動執行mysql請在/etc/rc.local中加入：

```
/usr/local/mysql/share/mysql/mysql.server start
```

建議您以後使用 mysql.php 來管理資料庫，這是一個可以從支援 PHP 的網頁上直接存取資料庫的程式。比起其他以 PHP 寫成的 MySQL 資料庫管理程式，我最喜歡這一個，因為只要將它放在網頁的目錄中，就可以執行了。只需一個檔案，完全不須做任何設定。你可以在本書第二片光碟 /examples 中取得。

10.2 安裝 apache

10.2.1使用 ports 安裝

我們要開始安裝 apache 了。如果您所要安裝的網頁伺服器只是要具備 apache 基本功能，您可以使用 ports 來安裝：

```
# cd /usr/ports/www/apache13
# make install
```

如果您要安裝 apache 並令其支援 ssl 及 php，可以使用下列指令：

```
# cd /usr/ports/www/apache13-ssl
# make install
# cd /usr/ports/www/mod_php4
# make install
```

但是這樣安裝出來的 PHP 並不能使用php 來繪圖，如果您需要功能更多的網頁伺服器，必須使用自以編譯的方法來安裝。

10.2.2 自行編譯

首先，在 /tmp 中建立一個目錄 work 並進入該目錄，取得以下檔案將它們放到該目錄下，以便管理，這些檔案在光碟二的 /ports/distfiles 目錄中都可以找到：

apache_1.3.23.tar.gz

mod_ssl-2.8.6-1.3.23.tar.gz

openssl-0.9.6b.tar.gz

mod_fastcgi_2.2.12.tar.gz

以下爲PHP所需的檔案，除了 php-4.1.2.tar.gz 外，其他的檔案是爲了要使 php 支援繪圖必須使用：

php-4.1.2.tar.gz

imap-2001.BETA.SNAP-0106191041.tar.gz

gd-1.8.4.tar.gz

zlib-1.1.4.tar.gz

t1lib-1.3.1.tgz

freetype2-2.0.6.tgz

jpeg-6b_1.tgz

png-1.2.1.tgz

1.解壓縮 apache：

```
# tar zxvf apache_1.3.23.tar.gz
```

2.安裝 openssl：

```
# tar zxvf openssl-0.9.6b.tar.gz
# cd openssl-0.9.6b
# ./config
# make
# make test
```

```
# make install
# cd ..
```

3.編譯 mod-ssl：

```
# tar zxvf mod_ssl-2.8.6-1.3.23.tar.gz
# cd mod_ssl-2.8.6-1.3.23
# ./configure --with-apache=../apache_1.3.23
# cd ..
```

4.先做一次 apache 的組態：

```
# cd apache_1.3.23
# ./configure --prefix=/usr/local/apache
# cd ..
```

5.安裝 PHP 之前先安裝 GD 及其所需檔案：

```
# tar zxvf zlib-1.1.4.tar.gz
# cd zlib-1.1.4
# make all install
# cd ..
# pkg_add -v jpeg-6b_1.tgz
# pkg_add -v png-1.2.1.tgz
# pkg_add -v t1lib-1.3.1.tgz
# pkg_add -v freetype2-2.0.6.tgz
# tar zxvf gd-1.8.4.tar.gz
```

```
# cd gd-1.8.4
# make install
# cd ..
# tar zxvf imap-2001.BETA.SNAP-0106191041.tar.gz
# cd imap-2001.BETA.SNAP-0106191041
# make bsf
# cd ..
# tar zxvf php-4.1.2.tar.gz
# cd php-4.1.2
# ./configure --with-mysql=/usr/local/mysql \
--with-apache=../apache_1.3.23 --enable-track-vars \
--with-imap=../imap-2001.BETA.SNAP-0106191041 \
--with-gd=/usr/local --enable-gd-native-ttf \
--with-t1lib \
--with-jpeg-dir=/usr/local --with-png-dir \
--with-freetype-dir \
--with-zlib-dir
# make
# make install
# cp php.ini-dist /usr/local/lib/php.ini
# cd ..
```

6.安裝 apache：

```
# cd apache_1.3.23/src/modules
```

```
# tar zxvf ../../../mod_fastcgi-2.2.12.tar.gz
# mv mod_fastcgi-2.2.12 fastcgi
# cd ../../
```

7. 設定 openssl 位置

設定 openssl 的目錄，如果您使用的 Shell 是使用 tcsh 的話：

```
# setenv SSL_BASE "../openssl-0.9.6b"
```

如果是使用 bash 的話：

```
# export SSL_BASE="../openssl-0.9.6b"
```

8.安裝 apache

```
# ./configure --prefix=/usr/local/apache \
--enable-shared=max \
--activate-module=src/modules/php4/libphp4.a \
--activate-module=src/modules/fastcgi/libfastcgi.a \
--enable-module=ssl --enable-suexec \
 --suexec-caller=nobody \
--suexec-docroot=/usr/local/apache/htdocs \
--suexec-userdir=public_html \
--suexec-logfile=/usr/local/apache/logs/suexec_log \
--suexec-uidmin=10 --suexec-gidmin=10
# make
# make certificate TYPE=dummy
```

```
# make install
# cd ..
```

大致上完成了，接著要設定 /usr/local/apache/conf/httpd.conf 來使 php 可以運作。找到 httpd.conf 中 php 的部份如下：

```
# And for PHP 4.x, use:
#
AddType application/x-httpd-php .php .phtml .php3
AddType application/x-httpd-php-source .phps
```

將 AddType 前的 # 拿掉並改成上面的樣子，存檔離開，如果沒有這一段文字則自行在 httpd.conf檔案最後自行加入。詳細的 httpd.conf 設定我們會在下一節說明。接下來執行以下指令來啓動 apache：

```
# /usr/local/apache/bin/apachectl start
```

啓動後，我們就可以使用瀏覽器連到該伺服器看看是否看得到網頁。如果可以的話，請使用 /usr/local/apache/bin/apachectl stop 來停止 apache 服務，再使用 /usr/local/apache/bin/apachectl startssl 來啓動具有 SSL 的 apache。並改以 ssl 連線到伺服器看看 https://your.server/ 來看是否成功。

我們接著要試試 php 可不可以運作。在 /usr/local/apache/htdocs/ 編輯一個檔名爲 test.php 的文字檔，內容如下：

```
<?
phpinfo();
?>
```

再使用瀏覽器連到該檔案，看看 php 是否正常：http://youserver/test.php

都完成之後，若開機即要執行 apache 的話，請在 /etc/rc.local 中加入：

```
/usr/local/apache/bin/apachectl startss/
```

10.2.3 後續系統設定

後續的設定就是要修改 /usr/local/apache/conf/httpd.conf 及 /usr/local/lib/php.ini，將它們依你的需要修改。休息一下，我們接著幾章便要說明這些檔案的設定。

10.3 http.conf 說明

/usr/local/apache/conf/httpd.conf 是 Apache 的主要設定檔。檔案中有 # 為開頭者是註解，用以說明設定的情形及方式，如果一行的開頭有 # 的話，該行對 Apache 就不會產生作用。全文可以分成三個部份，第一個區段是全域設定，用來設定 apache 執行時的重要設定。第二個部份是主要主機的設定，針對主要對外提供服務的主機加以設定。第三個部份是虛擬主機的設定，你可以在一台機器上設定多個 domain name 或多個 IP，並針對不同的 domain 來設定不同的目錄及相關參數。如果你有安裝 ssl，那麼還有第四個部份是 ssl 的設定。

修改完 httpd.conf 後，記得使用 /usr/local/apache/bin/apachectl restart 來重新啓動 Apache。

10.3.1 全域設定部份

```
# ServerType 可以設定為 inetd 或是 standalone。standalone 是採獨立常駐的方式，
# 即開機時就常駐於系統中。若是設定為inetd時，則是由inetd這個deamon來啓動
# 相關服務程式。一般來說以standalone的方式Server的效率會比較好，
#  除非您有特別的需要，否則建議以standalone的方式即可。
ServerType standalone
# apache 的根目錄，就是你安裝 apache 的目錄
ServerRoot "/usr/local/apache"

# 使用 NFS 時才會用到這項設定
#LockFile /usr/local/apache/logs/httpd.lock

# apache 啓動時會記錄 process id，並將它寫在下列設定的檔案中。
PidFile /usr/local/apache/logs/httpd.pid

# 設定 apache 程序的相關資訊記錄檔
ScoreBoardFile /usr/local/apache/logs/httpd.scoreboard

# 在 apache 1.3.6 版以前，還多了二個設定檔，就是 access.conf
# 及srm.conf，新版已經不需要了
#ResourceConfig conf/srm.conf
#AccessConfig conf/access.conf

# 設定和 client 幾秒內仍無法連上即切斷和 client 的連線
Timeout 300
```

```
# 可以設定為 On 或 Off，表示在完成 client 的連線要求後，是否要立即切斷連線。
# 一般會保持連線，以服務下一次的連線請求。如果設為 Off，每一次的連線要求
# 結束後，都會關閉連結，下一次的請求則要再開一個新的程序，這樣速度較慢。
# 除非你的硬體真的很差，每法同時有太多的程序，否則都設為 ON，以增加速度
KeepAlive On

# 同時保持連線要求的上限。如果你 KeepAlive 設為 On，這裡才會有作用。你可
# 以依照自己的備配提高這個值，以提高效能。如果設為 0 表示不限制
MaxKeepAliveRequests 100

# 設定持續連線時等待客戶端下一個請求的時間，超過此時間則視為連線中斷
KeepAliveTimeout 15

# Apache 會動態的依照系統系統的負載來調整所需的程序
# Apache 會定期檢查有多少連線要求在等待中
# 會在這個範圍中自動啟動適當數目的程序來等待請求
# 這裡的設定已滿足大多數網站的需求，你不必做更改
MinSpareServers 5
MaxSpareServers 10

# 如果你是以 standalone 的方式啟動 Apahce
# 這裡是設定啟動時要同時啟動多少個程序來等待連線請求
StartServers 5

# 同一時間可以連線的 client 數目，這個數目不應該太小
# 你可以依自己的硬體來調高這個值
MaxClients 150
```

```
# 每個請求子程序(child process)的最大數目。太多的子程序會佔用記憶體
# 及資源，如果設為 0 表示不限制
MaxRequestsPerChild 0

# 如果有設虛擬主機的話，你可以設定 Apache 要 listen 的IP或Port，當所架設的虛
# 擬主機若不在80埠號時，您就必須 在這指定其他埠號提醒Apache監看某個埠號。
#Listen 3000
#Listen 12.34.56.78:80

# 虛擬主機的相關設定，設定 Apache 可以接受連線請求的IP 位址或
# Domain Name，其設定值可以是 * 、 IP 位址或是完整的 Domain Name。
# BindAddress *

# Dynamic Shared Object (DSO) Support: 動態分享物件模組。
#可使 Apache在執行時直接加入某些需要的模組，使Apache在執行上更具彈性。
# 下列的各模組順序很重要，最好不要隨意更動
# Example:
# LoadModule foo_module libexec/mod_foo.so
LoadModule env_module       libexec/mod_env.so
.....略.....

# 是否要允許以 http://yourserver/server-status 來顯示
# 伺服器的設定狀態，預設是 Off。如果要設為 On，還必須
# 要設定下面 <Location /server-status> 的部份
#ExtendedStatus On
```

10.3.2 主要主機設定

```
### Section 2: 'Main' server configuration
# 如果你是以 inetd 的方式啟動 Apache，以下的部份設定並不會發生作用

# 設定 standalone 要傾聽的 port，如果 port 小於 1023
# 必須要以 root 才能啟動 Apache
Port 80

## SSL 支援
## 如果有使用 SSL，設定 SSL 要聽的 port
<IfDefine SSL>
Listen 80
Listen 443
</IfDefine>

# 如果你希望啟動 Apache 的使用者和跑 httpd 的使用者是不同人的話，
# 必須以 root 來啟動，這裡的設定就是你要使用的使用者名稱及其所屬群組。
# 所設的使用者必須是真的存在於系統千萬不要設成 root ，這對於安全上非常重要。
User nobody
Group nobody

#
# ServerAdmin: 你的信箱，這個信箱位址當網頁出現錯誤訊息時將出現在該頁面上
#
ServerAdmin www@mydomain.com
```




```
#
# ServerName 讓你可以設定一個主機名稱，該名稱和使用者連線
# 的名稱不同時，會傳回去給使用者。你可以設和你真實的主機
# 名稱不同，例如你可以在你的主機名稱前多加一個 www
#
# 請注意：你不能自己發明一個主機名稱並希望它可以運作，
# 這個主機名稱對你的機器而言，必須要是一個有效的 DNS name
# 如果你的主機並未擁有主機名稱，你可以使用 IP address
# 例如，123.45.67.89。如果你真實的 IP 都沒有，你可以使用127.0.0.1，
#  這是 TCP/IP local loop-back 的位址，通常用來作 localhost
#
ServerName www.alexwang.com

#
# DocumentRoot: 這個目錄是你放網頁的地方，你也可以放超連結
# 來將首頁指向其他地方。
DocumentRoot "/home/www"

#
# 所有 Apache 存取的目錄都可以對它們的屬性加以設定
#
# 首先我們先設定預設的權限
#
<Directory />
   Options FollowSymLinks
   AllowOverride None
</Directory>
```

```
#
# 這裡的設定是針對你網頁存放目錄及其子目錄
#
# 這裡可以使用的選項有 "All", "None", 或者是混合下列各項：
# "Indexes", "Includes","FollowSymLinks","ExecCGI",
# "MultiViews"
# Indexs 表示如果若找不到目錄中預設的首頁(DirectoryIndex
# 設定的檔案) 時，Apache 會自動產生index 列出目錄中的檔案。
# FollowSymlinks 表示允許符號鏈結（Symbolic Link）功能，
# 如果沒有此選項，Server會忽略系統中的連結檔案。
# Includes 允許SSI（Server Side Include）可以在該目錄下執行
# ExecCGI 允許執行CGI，若無此選項則該目錄中無法執行CGI程式
# MultiViews 允許內容協商的 MultiViews。
# None 關閉所有的選項，只允許Read。
# All 開啟所有的選項，除了MultiViews之外。
# 請注意，"MultiViews" 一定要明確寫出來，使用 "Options All"
# 並未包含 MultiViews。
#
<Directory "/home/www">
  Options Indexes FollowSymLinks MultiViews ExecCGI

#
# 這個選項是用來控制目錄中的 .htaccess 檔案可不可以覆蓋原本
# 對該目錄所設的權限。這個選項可以是 All 或是下列混合各項：
# "Options", "FileInfo", "AuthConfig", and "Limit"
# Options 允許該目錄位置在.htaccess檔中使用Options功能
# FileInfo 允許該目錄位置在.htaccess檔中使用AddEncoding、
# AddType、DefaultType、ErrorDocument等指令。
```

```
# AuthConfig 允許該目錄在.htaccess檔中使用AuthDBMGroupFile、
# AuthDBMUserFile， AuthGroupFile、AuthName、AuthType、
# AuthUserFile...等功能。
# Limit 允許使用Limit功能。
# None 停止.htaccess的功能。
# All 允許.htaccess所有功能。
#
  AllowOverride None

#
# 控制誰可以連到這個伺服器
# Order 表示先處理 allow 或是 deny。這裡是先 allow 再 deny
# 這裡 allow 跟 deny 的上下順序必須跟Order中所設定的順序
# 是一樣才可以。
# Allow from all 表示任何人都可以連到伺服器的這個目錄來瀏覽
# 若是設為Allow from freebsd.org 的話，則表示只有網域是
# 在freebsd.org 的人才可以連到該目錄中觀看。
# 當然若不想讓某些人連過來觀看這個目錄的網頁
# 可以加一行Deny from bad.Domain.com，當然您也可
# 以使用 IP 來代替 Domaine
#
  Order allow,deny
  Allow from all
</Directory>

#
# UserDir: 機器中的使用者預設放網頁的地方是在使用者
```

```
# 家目錄的那個目錄下使用者可以用
# http://www.hostname.com/~user 來連到 user 的首頁
#
<IfModule mod_userdir.c>
   UserDir public_html
</IfModule>

#
# 控制 UserDir 目錄的權限，和上述的差不多。
#<Directory /home/*/public_html>
#    AllowOverride FileInfo AuthConfig Limit
#    Options MultiViews SymLinksIfOwnerMatch IncludesNoExec
#    <Limit GET POST OPTIONS PROPFIND>
#        Order allow,deny
#        Allow from all
#    </Limit>
#    <LimitExcept GET POST OPTIONS PROPFIND>
#        Order deny,allow
#        Deny from all
#    </LimitExcept>
#</Directory>

#
# DirectoryIndex: 當連到目錄時，預設的網頁是哪一個
# 你可以用空白作為間隔，設定多個檔案，將依所設的順序尋找
# 最好加入 index.htm 及 index.html，如果有使用 PHP 的話
# 應該要再加入 index.php
```

```
#
<IfModule mod_dir.c>
   DirectoryIndex index.html index.php index.htm
</IfModule>


#
# AccessFileName: 目錄中放置控制資訊的檔案名稱
#
AccessFileName .htaccess


#
# 下列的設定是用來避免 .htaccess 檔案被客戶端使用者看到
# 既然 .htaccess 是用來控制權限的，你當然不會希望他被看到
# 另外，還有 .htpasswd 等檔案是用來控制該目錄密碼的，
# 這裡的設定也會影響其他 .ht 開頭的檔案
#
<Files ~ "^\.ht">
   Order allow,deny
   Deny from all
</Files>
#
# CacheNegotiatedDocs: 這個設定是用來告訴外面的 proxy
# 不要保留我的檔案，預設是註解掉，以降低流量
#
#CacheNegotiatedDocs


#
# UseCanonicalName: 使用這項設定以供 Apache 在需要時可以重
```

```
# 建自己的URL (回應該文件是從哪個 URL 出來的)。它將使用
# hostname:port 去回應要求，這個設定會影響 CGI 中的
# SERVER_NAME 及 SERVER_PORT
#
UseCanonicalName On

#
# TypesConfig 用來描述要去哪裡找 mine.types 的檔案
#
<IfModule mod_mime.c>
  TypesConfig /usr/local/apache/conf/mime.types
</IfModule>

#
# DefaultType 是網頁文件預設的 MIME type，你可以將它定義為
# "text/plain" 代表文件是 HTML 格式。如果你大多數的文件是
# binary 檔或是圖形，你可以使用 "application/octet-stream"
#
DefaultType text/plain

#
# mod_mime_magic module 使伺服器可以自己決定使用什麼 MIME type
# MIMEMagicFile 就是告訴該 module 要去哪裡找這些定義
# mod_mime_magic 並不是內定的 module，你可以重新編譯自行加入
# 這裡的 MIMEMagicFile 只有在包含了該模組後才有作用
#
<IfModule mod_mime_magic.c>
```




```
  MIMEMagicFile /usr/local/apache/conf/magic
</IfModule>

#
# HostnameLookups: 定義在 log 檔中要記錄 IP 或是 hostname
# 例如： www.apache.org (on) or 204.62.129.132 (off).
# 不要打開以節省向 DNS Server 要求解析的時間
#
HostnameLookups Off

#
# ErrorLog: 設定錯誤訊息的 log 要放在哪個檔案
# 如果你有設定在設定虛擬主機時另外指定它的 log 檔的話
# 該虛擬主機的 log 會在你所設定的地方
#
ErrorLog /var/log/apache_error_log

#
# LogLevel: 控制錯誤訊息的記錄等級
# 可以設為： debug, info, notice, warn,error,crit,alert,emerg
# debug一般用是在程式的除錯開發使用的
# info是指一般的資訊，notice指通知訊息
# warn指的是提示訊息，error指的是錯誤的訊息
# alert是指警告訊息，emerg則是緊急訊息
#
LogLevel warn

#
```

```
# 下列是自訂記錄檔格式，並取一個名字給它們
# 你可以在下列的設定中決定要記錄哪一種
# 這裡共取了四個名字，combin,common,referer,agent
# 四種都有不同的記錄內容，以 combin 最詳細
#
LogFormat "%h %l %u %t \"%r\" ...略...."" combin
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

#
# 設定有人存取檔案時要記錄哪些東西(依 common 的設定格式)
# 並指定要存放在哪裡
#
CustomLog /var/log/apache_access_log common

#
# 如果你想要再指定記錄其他東西的話，可以在這裡設定
#
#CustomLog /usr/local/apache/logs/referer_log referer
#CustomLog /usr/local/apache/logs/agent_log agent
#CustomLog /usr/local/apache/logs/access_log combined

#
# 設定是否將Server的版本和虛擬主機等資訊加入網頁中
# （通常出現在網頁錯誤或FTP列表時），Off表示關閉，
# EMail則是會把ServerAdmin的E-Mail連結加進去。
```

```
# 只能設下列三者之一: On | Off | EMail
#
ServerSignature On


#
# Aliases: 你可以再這裡設定任何的別名，格式是
# Alias fakename realname
#
<IfModule mod_alias.c>
    #
    # 請注意，如果你在別名(fakename)中加入了 / ，就必須在
    # URL 中表示出來。所以這裡 "/icons" 並未使用別名
    # 只有 "/icons/" 才有。如果 fakename 以 / 結束，則
    # realname 也要有，反之亦然。
    # 下面的 Options 等設定是該別名的目錄設定
    #
    Alias /icons/ "/usr/local/apache/icons/"

    <Directory "/usr/local/apache/icons">
        Options Indexes MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    #
```

```
    # ScriptAlias: 這用來控制 CGI 的連結
    # 注意需將Options的選項改為ExecCGI，才可以使用 CGI
    #
    ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"

    <Directory "/usr/local/apache/cgi-bin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>

</IfModule>
# End of aliases.



#
# 控制伺服器如何列出目錄內容
#
<IfModule mod_autoindex.c>

    #
    # FancyIndexing 可以美化列出的方式，也可以使用 standard
    #
    IndexOptions FancyIndexing


    #
```

```
# AddIcon* 控制如何依不同檔案格示顯示小圖式
# 是給 FancyIndexing 用的
#
AddIconByEncoding (CMP,/icons/compressed.gif) 略

AddIconByType (TXT,/icons/text.gif) text/*
.....略......
AddIcon /icons/blank.gif ^^BLANKICON^^

#
# DefaultIcon 是當不知格式時用的小圖式
#
DefaultIcon /icons/unknown.gif

#
# AddDescription 可以讓你對該檔案格式加以描述
# 格式： AddDescription "description" filename
#
#AddDescription "GZIP compressed document" .gz
#AddDescription "tar archive" .tar
#AddDescription "GZIP compressed tar archive" .tgz

#
# ReadmeName 是伺服器要去找的 README 檔案名稱
# 預設將列出在目錄的清單之後
#
# HeaderName 是要加在目錄清單前顯示的檔案名稱
```

```
    #
    # 如果在 Option 中有設 MultiViews 的話，伺服器會先去找
    # name.html 並包含它，如果該檔不存在就會去找 name.txt
    #
    ReadmeName README
    HeaderName HEADER

    #
    # IndexIgnore 是設定目錄中那個檔案不列出
    # 可以使用萬用字元 *
    #
    IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

</IfModule>
# End of indexing directives.

#
# 文件格式Document types.
#
<IfModule mod_mime.c>

    #
    # AddEncoding 設定編碼形式和所對應的副檔名。
    #
    AddEncoding x-compress Z
    AddEncoding x-gzip gz tgz

    #
```

```
# AddLanguage 讓你可以設定文件所要告知瀏覽器使用的語言，
#
# Danish (da) - Dutch (nl) - English (en) - Estonian (ee)
# French (fr) - German (de) - Greek-Modern (el)
# Italian (it) - Korean (kr) - Norwegian (no)
# Portugese (pt) - Luxembourgeois* (ltz)
# Spanish (es) - Swedish (sv) - Catalan (ca) -Czech(cz)
# Polish (pl) - Brazilian Portuguese (pt-br) -Japanese(ja)
# Russian (ru)
#
AddLanguage da .dk
....略.....
AddCharset UTF-8        .utf8

# LanguagePriority 設定使用語這的先後順序
#

<IfModule mod_negotiation.c>
LanguagePriority en tw da nl et fr de el
</IfModule>

#
# AddType 設定 PHP 所使用的 mime.types
#
# 裝 PHP3 的話就打開下列二行
#
#AddType application/x-httpd-php3 .php3
```

```
#AddType application/x-httpd-php3-source .phps
#
# 下面是給 PHP4 用的
#
AddType application/x-httpd-php .php .phtml .php3
AddType application/x-httpd-php-source .phps

AddType application/x-tar .tgz

#
# AddHandler 設定CGI-script和所對應的副檔名。若您會
# 使用到CGI程式，則必須將其註解拿掉打開其功能。
# 另外，有的CGI的副檔名會用到.pl，所以最好也加入。
#
# To use CGI scripts:
#
AddHandler cgi-script .cgi

#
# 設定是否加入有SSI功能的HTML和所對應的副檔名。
# 若您會使用到SSI的功能，則必須將其註解拿掉。
#
#AddType text/html .shtml
#AddHandler server-parsed .shtml

#
# Uncomment the following line to enable Apache's
# send-asis HTTP file feature
```

```
#
#AddHandler send-as-is asis

#
# If you wish to use server-parsed imagemap files, use
#
#AddHandler imap-file map

#
# To enable type maps, you might want to use
#
#AddHandler type-map var


</IfModule>
# End of document types.

#
# MetaDir: specifies the name of the directory in which
# Apache can find meta information files. These files
# >contain additional HTTP headers
# to include when sending the document
#
#MetaDir .web

#
# MetaSuffix: specifies the file name suffix for
# the file containing the meta information.
```

```
#
#MetaSuffix .meta

#
# 格式化網頁錯誤的訊息回應 (Apache style)
# these come in three flavors
#
#   1) plain text
#ErrorDocument 500 "The server made a boo boo.
# 注意 (") 的標誌並不會顯示
#
#   2) local redirects
ErrorDocument 404 /missing.html
# 設定當找不到網頁時就顯示 URL /missing.html
#ErrorDocument 404 /cgi-bin/missing_handler.pl
#
#   3) external redirects
#ErrorDocument 402 http://server.com/subscription_info.html

#
# 對各種瀏覽器作不同回應
#

<IfModule mod_setenvif.c>
  BrowserMatch "Mozilla/2" nokeepalive
  BrowserMatch "MSIE 4\.0b2;" nokeepalive 略
  BrowserMatch "RealPlayer 4\.0" force-response-1.0
  BrowserMatch "Java/1\.0" force-response-1.0
```

```
  BrowserMatch "JDK/1\.0" force-response-1.0

</IfModule>
# End of 對各種瀏覽器作不同回應
#
# 是否允許使用 http://servername/server-info 來顯示伺服器狀態
# 將 ".your_domain.com" 改成許可連結的網域
#
# <Location /server-status>
#    SetHandler server-status
#    Order deny,allow
#    Deny from all
#    Allow from .your_domain.com
# </Location>

#
# 是否允許使用 http://servername/server-info 來
# 顯示伺服器資訊 (必須要有載入mod_info.c)
# 將 ".your_domain.com" 改成許可連結的網域
# <Location /server-info>
#    SetHandler server-info
#    Order deny,allow
#    Deny from all
#    Allow from .your_domain.com
# </Location>

#
# 為防止Apache一個 1.1 版以前舊bug發生，用來將錯誤轉向處理並記錄的指令
```

```
# <Location /cgi-bin/phf*>
# Deny from all
# ErrorDocument 403 http://phf.apache.org/phf_abuse_log.cgi
# </Location>

#
# 是否使用 Apache 的 Proxy
# <IfModule mod_proxy.c>
#    ProxyRequests On
#...略....
#
# </IfModule>
# End of proxy directives.
```

10.3.3 虛擬主機及SSL的設定

```
### Section 3: Virtual Hosts
#
# VirtualHost: 你可以在一台機器上使用多個主機名稱
# 或 IP，並指定使用不同的目錄及設定
#

#
# 指定要使用的虛擬主機名稱或IP及port
#
#NameVirtualHost *

#
# VirtualHost 範例
# 第一個設定是當未知主機名稱時用的
#<VirtualHost *>
#  ServerAdmin webmaster@dummy-host.example.com
#  DocumentRoot /www/docs/dummy-host.example.com
#  ServerName dummy-host.example.com
#  ErrorLog logs/dummy-host.example.com-error_log
#  CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>
# SSL 設定
## 我將省略大多數的說明
## 只寫我們要改的地方
## SSL Global Context
```

```
##
## All SSL configuration in this context applies both to
## the main server and all SSL-enabled virtual hosts.
##
....略....
##
## SSL Virtual Host Context
##
# General setup for the virtual host

<VirtualHost _default_:443>

# 在這裡改你的網頁位址及 log 檔位址即可
#
DocumentRoot "/home/www"
ServerName www.alexwang.com
ServerAdmin jack@myserver.com
ErrorLog /var/log/apache_error_log
TransferLog /var/log/apache_access_log
</VirtualHost>
```

10.4 php.ini 說明

/usr/local/lib/php.ini 是 PHP 的設定檔，檔案的格式是以 ";" 為註解、以 [ ] 包起來的是區段的名稱，二種都不會代表任何意義。在安裝完 PHP 時，我們從 PHP 原始碼中複製一份 php.ini-dist 在 /usr/local/lib/php.ini，這樣我們才可以針對 PHP 來調整成我們要的參數。 修改完後，也要重跑 Apache 才可以有作用。

一般而言，我們不太需要更改這些設定，但如果你想要對它有更深入的了解，我們會一一說明：

```
[PHP]
; $Id: php.ini-dist,v 1.78.2.2 2001/06/01 03:20:49 sniper Exp $


;;;;;;;;;;;;;;;;;;;
; 關於這個檔案 ;
;;;;;;;;;;;;;;;;;;;
;
; 在這裡設定的參數名稱有大小寫之分
; 例如 - foo=bar 和 FOO=bar 所代表的意義不同
;
; 所設定的值可以是字串、數字、PHP 的常數 (如 E_ALL 或 M_PI)
; INI 常數 (On, Off, True, False, Yes, No and None) 或是
; 一個運算表示式 (如 E_ALL & ~E_NOTICE), 或是引號內的字串 ("foo")
;
; 運算表示式在 INI 檔中只能使用下列符號及運算子
; | bitwise OR
; & bitwise AND
```

```
; ~ bitwise NOT
; ! boolean NOT
;
; 布林運算 (Boolean) 可以使用下列的值作為真：1, On, True 或是 Yes
; 也可以使用下列的值作為假： 0, Off, False 或 No
;
; 如果要指一個空字串，只可在等號後什麼都不加，或是以 none 表示。
;
; foo = ; 將變數 foo 設為空字串
; foo = none ; 將變數 foo 設為空字串
; foo = "none" ; 將變數 foo 設成字串 'none'
;
; 你果你要動態載入一些表示式所方的變數 (可能是 PHP extension
; 或是 Zend extension)，你必須在載入後才能使用該變數
;
; 所有 php.ini-dist 的設定都是內建的預設值，如果沒有 php.ini 時
; 或者當你刪除該行，就會使用內建的預設值

;;;;;;;;;;;;;;;;;;;;
; 程式語言選項    ;
;;;;;;;;;;;;;;;;;;;;

; 是否要在 Apache 中啓動 php 引擎
engine = On

; 可以使用 <? 的標籤，不然的話，只能使用 <?php 和 <script>
short_open_tag = On
; 是否允許 ASP 格式的標籤 <% %>
```

```
asp_tags = Off

; 使用浮點數 (floating point numbers) 要用多少數字
precision = 14

; Enforce year 2000 compliance (在某些瀏覽器可能會產生問題)
y2k_compliance = Off

; output buffering 可以讓你就算已經送出 body content 後，還
; 可以再送 header (包括 cookies)，只是這樣會減慢一點 php 輸
; 出的速度。你也可以在執行程式時呼叫 output buffering 的函式
; 來取得這項功能。或者就在這裡設成 On 來啓動吧。
; 一般我會設成 Off，只有當你很常用到這個功能才設成 On。
output_buffering = Off

; 你可以將你的 php 程式輸出轉向到一個函式，例如，如果你
; 將 output_handler 設為 "ob_gzhandler", 輸出將會使用 gzip
; 壓縮網頁給瀏覽器
output_handler =

; 壓縮輸出要使用 zlib 函式庫，這裡可以使用的值可以是
; 'off', 'on', 或者是用在壓縮的暫存區大小 (預設是 4KB)
zlib.output_compression = Off

; Implicit flush 告訴 PHP 每次輸出一個區段都要 flush 強制
; 將暫存區的東西輸出給瀏覽器。這和在 PHP 程式在 print() 或
; echo() 之後呼叫 flush() 函式有相同的效果。這個選項最好不
; 要打開，否則效率差很多，只有用於除錯時才會打開他。
```




```
implicit_flush = Off

; 是否要強制在呼叫變數時都使用傳址呼叫，這個功能未來版本的
; PHP/Zend 將取消。比較好的方式是在函式定義時就宣告變數的
; 呼叫方式以傳址呼叫。你可以在這裡將它設為 Off，來看你所寫
; 的程式是否可以在未來版本的 PHP 執行，而參數在傳遞時就會以
; 值，而非在記憶體位址。
allow_call_time_pass_reference = On


;
; 安全模式
;
safe_mode = Off

safe_mode_exec_dir =

; 設定一些環境變數可能造成安全性的破壞
; 這些變數如 comma-delimited list of prefixes。在安全模
; 式，使用著只能使用所定義的前綴字串作起始的變數，預設
; 只有以 PHP_ 作開始的變數 (如 PHP_FOO=BAR)
;
; 請注意：如果設為空字串，PHP 允許使用者設任何環境變數
safe_mode_allowed_env_vars = PHP_

; 這個指令包含了使用者不能以 putenv() 改變的環境變數
; 的 comma-delimited list ，這個變數可以所設定的保護
; 就算 safe_mode_allowed_env_vars 設定允許也不能改變
```

```
safe_mode_protected_env_vars = LD_LIBRARY_PATH

; 這個設定可以讓你因為安全的理由而取消一些函式
; 不管你的 Safe Mode 是設為 On 或 Off 都不會影響它
disable_functions =

; 在 Highlighting mode 所要使用的符號顏色
; 只要是在 <font color="??????"> 中的東西都可以
highlight.string = #CC0000
highlight.comment = #FF9900
highlight.keyword = #006600
highlight.bg = #FFFFFF
highlight.default = #0000CC
highlight.html = #000000


;
; 其他設定
;
; 決定是否要在伺服器上使用 PHP
expose_php = On


;;;;;;;;;;;;;;;;;;;
; 資源限制    ;
;;;;;;;;;;;;;;;;;;;

max_execution_time = 30 ; 每個 PHP 程式最大的執行時間
```

```
memory_limit = 8M ; 每個 PHP 程式最大可以消耗多少記憶體 (8MB)


;;;;;;;;;;;;;;;;;;;;;;;;;;;
;   錯誤的處理及記錄   ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;

; error_reporting 可以讓你設定要回報的錯誤內容
; E_ALL - All errors and warnings，所有錯誤及警告
; E_ERROR - fatal run-time errors，執行時的致命錯誤
; E_WARNING - run-time warnings (non-fatal errors)，執行時的警告
; E_PARSE - compile-time parse errors
; E_NOTICE - run-time notices (這個警告通常是你的程式碼有問題
; 或者也有可能是內部錯誤 (例如使用一個未初始化的變數)
; E_CORE_ERROR - 當 PHP 起始時的 fatal errors
; E_CORE_WARNING - 當 PHP 起始時的 warnings (non-fatal errors)
; E_COMPILE_ERROR - fatal compile-time errors
; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
; E_USER_ERROR - user-generated error message
; E_USER_WARNING - user-generated warning message
; E_USER_NOTICE - user-generated notice message
;
; 範例：
;
; - 除了 notice 外顯示所有錯誤
;
;error_reporting = E_ALL & ~E_NOTICE
```

```
;
; - 只顯示 errors
;
;error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
;
; - 除了 notice 外顯示所有錯誤
;
error_reporting = E_ALL & ~E_NOTICE

; 將錯誤顯示在輸出的頁面上， 如果是輸出網頁，建議你把這個功能
; 關掉，並以 error logging 將它記錄在檔案中。否則顯示一些錯誤
; 在網頁上可能會有潛在的安全問題，如檔案位置、資料庫的輪廓或是
; 一些其他的資訊
display_errors = On

; 就算打開了 display_errors，在起始 PHP 時發生的錯誤並不會顯示
; 建議你除了除錯外不要打開這個功能
display_startup_errors = Off

; 將錯誤記錄在檔案中，建議你將網頁產生的錯誤記錄下來
; 這個打開後，將會記錄在你 Apache 的 error_log 檔中
;log_errors = Off
log_errors = On

; 是否要將最後的錯誤訊息存在 $php_errormsg 的變數中
track_errors = Off

; 在輸出錯誤訊息前要先輸出什麼字串，
```

```
; 可以用來改變網頁中字的顏色
;error_prepend_string = "<font color=ff0000>"

; 在輸出錯訊息後要輸出什麼字串
;error_append_string = "</font>"

; 要將 log 記錄在那個檔案
;error_log = filename
; Log errors to syslog (Event Log on NT, not valid in Windows 95).
;error_log = syslog

; 如果在字串上使用 + 的運算是否要警告
warn_plus_overloading = Off



;;;;;;;;;;;;;;;;;;;
; 資料處理 ;
;;;;;;;;;;;;;;;;;;;
;
; 請注意 - track_vars 在 PHP 4.0.3 中永遠有效
; PHP 如何處理輸出給 URL 多個變數，用什麼字
; 串將它們分開，預設是 "&"
;arg_separator.output = "&amp;"

; PHP 如何處理從 URL 傳來的多個變數，用什麼字
; 串將它們分開，預設是 "&"
; 請注意：這裡設中字的任可一個 "字元" 都將視為一個分開的符號
```

```
;arg_separator.input = ";&"

; 這裡的設定是當 PHP 接收來自 GET, POST, Cookie,環境變數(
; Environment) 和 內建變數 (Built-in variables)有重複時，
; 要處理的順序。以 (G, P, C, E & S 表示上述的方法，可以寫成
; EGPCS 或 GPC). 處理的順序是由左至右，當變數名稱相同時，
; 比較慢處理的將覆蓋舊的值。
variables_order = "EGPCS"

; 是否要將 EGPCS 的變數註冊成全域變數
; 這個當你在以 $HTTP_*_VARS[] 處理 GPC 變數時就有作用
;
; 最好不要在你的程式中預設 register_globals 打開的，
; 全域變數沒有使用好可能會有安全問題
register_globals = On

; 這是要告訴 PHP 是否要註冊 argv&argc 變數 (這包含 GET 的資訊)
; 如果你不使用它，則可以關掉以增加處理效率
register_argc_argv = On

; 設定 PHP 在接收 POST 資料時最大的容量大小
post_max_size = 8M

; 這個選項已沒有作用，請使用 variables_order 替代它
gpc_order = "GPC"

; Magic quotes
;
```

```
; GET/POST/Cookie 進來時使用 Magic quotes
magic_quotes_gpc = On

; Magic quotes for runtime-generated data,
; 如 data from SQL, from exec(), 等
magic_quotes_runtime = Off
; Use Sybase-style magic quotes (escape ' with '' instead of \').
magic_quotes_sybase = Off

; 自動在所有 PHP 檔案之前或之後包含一個檔案
auto_prepend_file =
auto_append_file =

; PHP 4.0b4 會送出一個語言編碼方式 MIME type
; 如果要把內定使用語言取消，就把 charset 設為空字串
;
; PHP 預設使用的 MIME type 是 text/html
default_mimetype = "text/html"
;default_charset = "iso-8859-1"


;;;;;;;;;;;;;;;;
; 路徑及目錄 ;
;;;;;;;;;;;;;;;;

; UNIX: "/path1:/path2"
;include_path = ".:/php/includes"
```




```
;
; Windows: "\path1;\path2"
;include_path = ".;c:\php\includes"

; PHP 文件的位置，只有非空字串才有作用
doc_root =
; 當 php 以 /~usernamem 打開文件時，所要使用的目錄
; 就是使用者目錄中 PHP 文件要放在哪裡
; 只有非空字串才有作用
user_dir =

; 其他可以載入的模組位置
extension_dir = ./

; 是否要使用 dl() 函式。dl() 在一些多重執行緒的伺服器
; 可能不會運作，如 IIS 或 Zeus，這時它會自動取消 dl()
enable_dl = On

;;;;;;;;;;;;;;;;
;  檔案上傳  ;
;;;;;;;;;;;;;;;;

; 是否要使用 HTTP 上傳檔案
file_uploads = On

; 上傳檔案時所要使用的暫存目錄，如果沒有指定就會使用系
; 統內定的暫存目錄
;upload_tmp_dir =
```

```
; 最大上傳檔案大小，我改成 5MB，一首歌都大於原本的 3MB
upload_max_filesize = 5M

;;;;;;;;;;;;;;;;;;
; Fopen wrappers ;
;;;;;;;;;;;;;;;;;;

; 是否允許使用 URL (如 http:// 或 ftp://) 做為 fopen() 所
; 要開啓的檔案
allow_url_fopen = On

; 定義可匿名的 FTP 所要使用的密碼
;from="john@doe.com"

;;;;;;;;;;;;;;;;;;
; 動態延伸模組 ;
;;;;;;;;;;;;;;;;;;
;
; 如果你希望自動載入延伸模組，請使用下列設定
;
; extension=modulename.extension
;
; 例如在 windows 下，使用：
;
; extension=msql.dll
;
; ... 或在 UNIX:
```

```
;
; extension=msql.so
;
; 請注意，這裡只能用模組名稱，不能包含目錄
; 你要先在上面的 extension_dir 設定模組的目錄


;Windows Extensions
;Note that MySQL and ODBC support is now built in,
;so no dll is needed for it.
;
;extension=php_bz2.dll
;.... 略....
;extension=php_zlib.dll

;;;;;;;;;;;;;;
; 模組設定 ;
;;;;;;;;;;;;;;

[Syslog]
; 是否要定義 syslog 變數，如 $LOG_PID, $LOG_CRON 等。
; 關掉它會有比較好的執行效率
; 你可以在程式中使用 define_syslog_variables() 來定義它們
define_syslog_variables = Off

[mail function]
; 只用於 Win32
SMTP = localhost
```

```
; 只用於 Win32
sendmail_from = me@localhost.com

; 只用於 Unix，你也可以加入參數 (內定值: 'sendmail -t -i').
;sendmail_path =

[Logging]
; 這個設定用於 example logging ，請參考 examples/README.logging
;logging.method = db
;logging.directory = /path/to/log/directory

[Java]
;java.class.path = .\php_java.jar
;java.home = c:\jdk
;java.library = c:\jdk\jre\bin\hotspot\jvm.dll
;java.library.path = .\

[SQL]
sql.safe_mode = Off

[ODBC]
;...略...

[MySQL]
; 讓你可以做持續的連結資料庫
mysql.allow_persistent = On
```

```
; 最大的持續連結， -1 表示不限
mysql.max_persistent = -1

; 最大的連結 (持續連結 + 非持續連結)。 -1 表示不限
mysql.max_links = -1

; mysql_connect() 預設使用的 port。如困沒有設定，
; mysql_connect() 將使用 $MYSQL_TCP_PORT 或者是 mysql
; 在 /etc/services 中的設定或是在安裝編輯時所設的 MYSQL_PORT
mysql.default_port =

; MySQL 作本地連結時內定使用的 socket name
; 如果沒有設定，將以 MySQL 預設為主
mysql.default_socket =

; mysql_connect() 內定所使用的主機 (在安全模式中沒有作用)
mysql.default_host =

; mysql_connect() 內定的使用者 (在安全模式中沒有作用)
mysql.default_user =

; mysql_connect() 內定的密碼 (在安全模式中沒有作用)
; 請注意，將密碼存在這個檔中並不是一個好的方法
; *任何* PHP 程式都可以經由
; 'echo cfg_get_var("mysql.default_password") 來取得密碼
; 所有使用者都將知道密碼
mysql.default_password =
```

```
[mSQL]
; 允許使用持續的連結資料庫
msql.allow_persistent = On

;...以各種資料庫的設定都和 MySQL 差不多，故略...

[Session]
; Handler 所使用儲存及取得的資料
session.save_handler = files

; 傳遞給 save_handler 的參數。這是 session 將存資訊的目錄
session.save_path = /tmp

; 是否要使用 cookies
session.use_cookies = 1

; session 的名稱(用來作 cookies 名稱)
session.name = PHPSESSID

; 啟動時是否要重設 session
session.auto_start = 0

; cookie 要存在幾秒，如果是 0，代表直到重新啟動瀏覽器
session.cookie_lifetime = 600

;  cookie is 的有效路徑
session.cookie_path = /
```

```
; cookie 的主機來源
session.cookie_domain =

; Handler 使用的 serialize data.
; php 是 PHP 標準使用的 serialize data
session.serialize_handler = php

; 是否要在 session 重設時啟動'garbage collection'
session.gc_probability = 1

; 幾秒後 session 資料將被視為垃圾 'garbage' 並回收
session.gc_maxlifetime = 1440

; 檢查 HTTP Referer 來使外部所在的 URLs containing ids 無效
session.referer_check =


; 要從檔案中讀多少 bytes
session.entropy_length = 0


; 在這裡指定要建立的 session id
session.entropy_file =

;session.entropy_length = 16

;session.entropy_file = /dev/urandom

; Set to {nocache,private,public} to
;determine HTTP caching aspects.
session.cache_limiter = nocache

; 存在暫存區中的 session 文件幾分鐘後到期
```

```
session.cache_expire = 180

; 支援短暫的 sid support 來相容 --enable-trans-sid.
session.use_trans_sid = 1

url_rewriter.tags =
  "a=href,area=href,frame=src,input=src,form=fakeentry"

[MSSQL]
; Allow or prevent persistent links.
mssql.allow_persistent = On
; ...MSSQL 和之前的資料庫差不多，故略...
; ...以下略

; Local Variables:
; tab-width: 4
; End:
```

10.5 .htaccess 應用

Apache 允許使用者在目錄下放置一個檔案 來控制該目錄的存取權限。預設是使用 .htaccess 這個檔。你可以自行用文書軟體編輯一個檔名爲 .htaccess 的檔案來設定檔案所在目錄的權限。 不過也要看 httpd.conf 中關於該目錄的 AllowOverride 是否有打開，如果有打開才可以用 .htaccess 的檔案去覆蓋原本對該目錄的設定。

首先先編輯 /usr/local/apache/conf/httpd.conf，在網頁目錄設定的區段

```
<Directory "/home/www">
  Options Indexes FollowSymLinks MultiViews ExecCGI
  AllowOverride AuthConfig
  Order allow,deny
  Allow from all
</Directory>
```

設定 AllowOverride 的部份，如果設爲 None 表示不允許使用者變更目錄設定，設爲 AuthConfig 表示可以使用 AuthDBMGroupFile、AuthDBMUserFile， AuthGroupFile、AuthName、AuthType 等認證的功能。所以我們設定爲 AuthConfig。

如果 Directory 的區段中，AllowOverride 是設成 All，你就可以在 .htaccess 檔案中設定所有選項，如 Options, AllowOverride 等。

實際應用

.htaccess 最常用的一個例子是用來將目錄設定需認證才能讀取。假設你要將某個目錄設定需要密碼才能讀取，你可以在該目錄下編輯一個名爲 .htaccess 的文字檔，內容如下：

```
AuthName "管理專區"
AuthType "Basic"
AuthUserFile "/var/adminDir.pw"
require valid-user
```

其中請注意各參數的大小寫。這裡我們設定儲存使用者帳號及密碼的檔案是 /var/adminDir.pw。

接著使用指令：

```
# /usr/local/apache/bin/htpasswd -c /var/adminDir.pw username
New password: 輸入 username 的密碼
Re-type new password: 再輸入一次
```

來建立檔案 /var/adminDir.pw 並加入使用者username，日後要再新增使用者不必加參數 -c。

接著使用瀏覽器連到該目錄時，將出現圖 10-1：

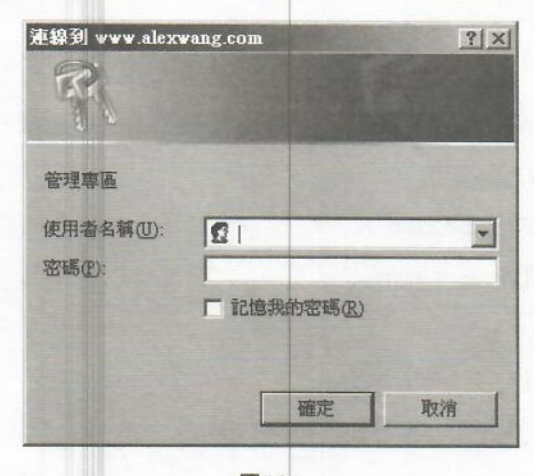


圖 10-1

此時輸入你設定的 username 及密碼即可。

10.6 虛擬主機

我們可以在一台機器上設定多個主機名稱或 IP，並依不同名稱來決定其根目錄所在。當使用者連線到我們的主機時，每一個不同的名稱所看到的根目錄都不同。

要達到這樣的功能，我們必須先確定主機有多個 DNS 名稱，這樣別人打該主機名稱才會對應到你的 IP。 我們先來說在一台主機上使用多個 DNS 的範例。

假設你的主機 IP 是 123.456.78.9 ，上面有二個主機名稱，一個是 www.abc.com，另一個是 www.cde.net。也就是說不管是使用上述哪一個 Domain Name，都可以連到 123.456.78.9。接著請編輯 httpd.conf，在虛擬主機的部份加入下列設定：

```
# 設定本機所使用的 IP
NameVirtualHost 123.456.78.9

# 設定 www.abc.com 的管理者帳號、存放網頁的目錄及log 所在
<VirtualHost 123.456.78.9>
    ServerAdmin jack@abc.com
    DocumentRoot /home/www/abc
    ServerName www.abc.com
    ErrorLog /var/log/abc_error_log
    CustomLog /var/log/abc_access_log common
</VirtualHost>
# 設定 www.cde.com 的資料
<VirtualHost 123.456.78.9>
```

```
    ServerAdmin tom@cde.com
    DocumentRoot /home/www/cde
    ServerName www.cde.com
    ErrorLog /var/log/cde_error_log
    CustomLog /var/log/cde_access_log common
</VirtualHost>
```

做完上面的設定後，就可以使用 /usr/local/apache/bin/apachectl restart 重新啓動 Apache 了。如果你還有別的 Domain Name 指向 123.456.78.9 的話，例如 www.fgh.org ，但你並未設定其 Virtual Host 資料，Apache 將以第一個設定的 Virtual Host 資料爲主。在這個範例裡，當你打 www.fgh.org 會連到 www.abc.com 的設定。

必須要注意的是，有些客戶端的連線軟體並不支援 Name-Based 的虛擬主機，要支援 name-based virtual host，客戶端必須送出 HTTP 的標頭，也就是瀏覽器必須支援 HTTP/1.1。請放心，我們常用的 IE、Netscape、lynx 都有支援。

10.7 MRTG 流量分析

如果您想要知道網站流量的使用情形，我們可以安裝 MRTG 這套軟體經由網頁來監看網路流量。MRTG 會去收取 SNMP (Simple Network Management Protocol) 所產生的資料，因此所要記錄的機器必須要安裝 SNMP。在你的主機上安裝 MRTG 後，你不僅可以收集自己的流量資料，也可以收集區域網路上其他可以接收到的 SNMP 資料。

10.7.1 安裝 SNMP

我們使用 ports 來安裝 SNMP：

```
# cd /usr/ports/net/net-snmp
# make install clean
```

編譯了一陣子之後會出現要我們按任意鍵繼續：

```
-Press return to continue-
System Contact Information (root@): 輸入你的 E-mail
System Location (Unknown):          輸入機器所在位置
Location to write logfile (/var/log/snmpd.log): 輸入 log 檔位置
Location to write persistent information (/var/ucd-snmp): 要存放 snmp.conf 的目錄
```

安裝完成後執行 /usr/local/etc/rc.d/snmpd.sh start 來啓動 snmpd，如果出現錯誤則再進入下列步驟：

cd /usr/local/etc/rc.d/，將 snmpd.sh 原本的內容刪除，並加入下列這一行：

```
/usr/local/sbin/snmpd
```

接著再執行 /usr/local/etc/rc.d/snmpd.sh 便可啓動 SNMP。

10.7.2 安裝 MRTG

我們使用 ports 來安裝 MRTG：

```
# cd /usr/ports/net/mrtg
# make install
```

接著要產生 MRTG 的設定檔

```
# cd /usr/local/etc/mrtg
# rehash
# cfgmaker public@alexwang.com >mrtg.cfg
```

這裡的 public 是 community name，預設是 public，如果你使用錯的 community name，你可能會從要記錄的設備上得到錯誤回應。而 alexwang.com 是你所要記錄的主機位置。mrtg.cfg 就是所要產生的設定檔名。

如果您想要記錄多個主機，只要在 cfgmaker 時多加入主機名稱即可，例如：

```
# cfgmaker public@alexwang.com public@dns1.alexwang.com >mrtg.cfg
```

這樣就會同時記錄上面二台主機的流量了。

產生基本的設定檔後，我們可以再編輯剛才產生的設定檔 ee mrtg.cfg，在檔案開頭的部份加入一些客製化的設定：

```
# 如果要使用中文的 MRTG 則加入下面這一行
Language: big5
```

```
# 設定你的 MRTG 要放在哪個目錄，應該要放在網頁可以
# 連結到的地方，我的網頁根目錄是 /home/www，所以我將
# MRTG 放在下面的目錄。
WorkDIR:/home/www/mrtg

# 預設的 MRTG 所產生的圖時間是由右到左
# 我喜歡由左到右，故加入下面這一行
Options[_]: growright
```

接著請建立一個你在 mrtg.cfg 中設定的 WorkDIR 的目錄：

```
# mkdir /home/www/mrtg
```

然後使用指令 indexmaker 來建立 MRTG 的首頁：

```
#indexmaker -title '流量統計' - r '.' -output /home/www
/mrtg/index.html mrtg.cfg
```

這裡的參數 -title 是該 index.html 檔的 title，-r 是 index.html 要去找圖的相對位置，而 -output 就是要輸出的檔案位置，預設是stdio(通常指的是螢幕)。

輸出的檔案 index.html 你也可以使用其他的網頁編輯軟體再去修改美化它。接下來要將MRTG的一些圖片檔複製到 mrtg 的目錄裡：

```
# cd /usr/ports/net/mrtg/work/mrtg*
# cd images
# cp * /home/www/mrtg/
```

最後啓動 mrtg：

```
# /usr/local/bin/mrtg /usr/local/etc/mrtg/mrtg.cfg
```

第一次執行上面的指令可能會有一些錯誤訊息，不要理它，再多執行幾次就好了。沒問題之後，使用指令 crontab -e 來把上述指令每 5 分鐘執行一次，加入下面這一行：

```
5,10,15,20,25,30,35,40,45,50,55 * * * * /usr/local/bin/mrtg /usr/local/etc/mrtg/mrtg.cfg
```

現在你可以使用 http://yourserver/mrtg 來連去看看。最後別忘了移除安裝過程的暫存檔：

```
# cd /usr/ports/net/mrtg/
# make clean
```

10.8 伺服器管理

10.8.1 apachectl

這是一個管理 Apache Server 的工具。

參數	說明
start	啓動 Apache Server
stop	停止 Apache 服務
restart	重新啓動 Apache
startssl	啓動具 SSL 功能的 Apache Server

範例： /usr/local/apache/bin/apachectl startssl

10.8.2 ab

這是用來測試 Apache 效能的工具。你可以針對某個 URL 來模擬出連續的連線請求 (不限本地主機)，並設定同時間要模擬多少連線。

參數	說明
-n requests	要做多少次連線請求，requests 爲次數
-c concurrency	同時有多少個連線，concurrency 爲個數
-t timelimit	最多等待回應的秒數
-p postfile	要以 POST 方法連線所欲送出的參數檔案。postfile 爲存放參數的檔案名稱。

例如，我要對自己的機器中的 /cgi-bin/test.cgi 作測試，模擬 1000 次請求，每次最多同時 20 個連線，只要在命令列執行指令：

```
# /usr/local/apache/bin/ab -n 1000 -c 20 http://127.0.0.1/cgi-bin/test.cgi
```

等了幾秒之後出現：

```
This is ApacheBench, Version 1.3d <$Revision:1.58 $> apache-1.3
Copyright (c) 1996 Adam Twiss, Zeus Technology Ltd,
Copyright (c) 1998-2001 The Apache Group,http://www.apache.org/

Benchmarking 127.0.0.1 (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
```

```
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Finished 1000 requests
Server Software:      Apache/1.3.22
Server Hostname:      127.0.0.1
Server Port:          80

Document Path:        /cgi-bin/test
Document Length:      18307 bytes

Concurrency Level:    20
Time taken for tests: 51.911 seconds
Complete requests:    1000
Failed requests:      0
Broken pipe errors:   0
Total transferred:    18520000 bytes
HTML transferred:     18307000 bytes
Requests per second:  19.26 [#/sec] (mean)
Time per request:     1038.22 [ms] (mean)
Time per request: 51.91 [ms] (mean, across all cont. requests)
Transfer rate:        356.76 [Kbytes/sec] received

Connnection Times (ms)
          min  mean[+/-sd] median   max
Connect:     0    7  38.2     2   329
Processing: 338 1022 133.9  1003  1774
```

```
Waiting:    223  1011  136.5   993  1774
Total:      338  1029  121.7   1006  1774

Percentage of the requests served within a certain time (ms)
  50%   1006
  66%   1026
  75%   1051
  80%   1074
  90%   1175
  95%   1265
  98%   1347
  99%   1545
 100%   1774 (last request)
```

您可以增加最多同時連線數目及連線次數，操看看你機器的上限在哪裡。

還有更多的參數，詳細用法請 man -M /usr/local/apache/man ab。

10.8.3 壓縮備份 log 檔

隨著使用人數的增加，網站的 log 檔可能會越來越大，我們可以使用 FreeBSD 內定的 newsyslog 來把舊的 log 備份起來。在 newsyslog 中，我們可以指定要備份多少個 log 檔，超過之後會自動刪除最舊的檔案。

首先編輯 /etc/newsyslog.conf 加入下列二行：

```
/var/log/apache_access_log    644  7    *   $W0D1  Z
/var/log/apache_error_log     644  7    *   $W0D2  Z
```

以上二行的意義是將 /var/log/apache_access_log 這個檔案做備份，備份後的檔案名稱像這樣 apache_access_log.0.gz 。備份後該檔案的權限是 644，最大的數字到 7，也就是最多八個檔案，不限制檔案多大時要備份，選在每週日半夜 1 點時備份，並將該檔以 gzip 壓縮。

chapter 11

郵件伺服器

11.1 概論

這裡我們將介紹使用 FreeBSD 來作爲 Mail Server。做爲一台 Mail Server，我們要設定的是最少要做到可以正常使用 POP3 及 SMTP 來讓使用者收發信。

傳統上，SMTP 在接受使用者寄信時，並不須經過身份認証，任何人都可以使用你的主機來製造垃圾信。因此 FreeBSD 內定的 Sendmail 是不接受 SMTP 寄信的。而一般的 ISP 業者大多是以控制連線來源的方式，禁止非允許網域的使用者 RELAY。但如果我們以控制連線來源的方式，便無法在其他非允許的IP位址使用SMTP，這對於想要任何地方都可以發信的人十分不分便。因此，本章裡我們將介紹如何讓使用者透過 FreeBSD 使用 SMTP 身份認證的功能來寄信，讓要使用諸如 outlook 以 SMTP 寄信的使用者必須先通過本機的身份認證。

另外，你也無法在別的電腦使用收信軟體來經由 POP3 收信，因爲預設並未裝設任何 POP3 軟體。

所以我們這裡要做的就是設定 sendmail 成爲可以使用 SMTP 來寄信，但又可以兼顧避免成爲垃圾中繼站。再來是安裝 POP3 軟體，設使用者可以經由 POP3 協定來收信。

11.2 具身份認證的 sendmail

本文參考自中央研究院計算機中心張毓麟先生所發表的「具身分認證的郵件傳送系統」，該文件的網址是http://beta.wsl.sinica.edu.tw/~ylchang/Email/sendmail-auth。張先生對於在 FreeBSD 系統上建立安全的郵件伺服器有多篇文章，您可以自http://beta.wsl.sinica.edu.tw/~ylchang/Email/index.html 取得更多內容。

11.2.1 安裝 Cyrus SASL

SASL (Simple Authentication and Security Layer) 可以讓一些通訊協定(例如 SMTP、IMAP 等) 具有身份認證的功能。Sendmail 自從 8.10 就支援 SASL 的功能。目前 cyrus-sasl 版本是 cyrus-sasl-1.5.27.tar.gz，我們可以 ftp 到各大 FTP 站台的 distfiles 目錄下去取得最新版的 cyrus-sasl或從本書光碟中取得。取回後以下列指令安裝。

```
# tar zxf cyrus-sasl-1.5.27.tar.gz
# cd cyrus-sasl-1.5.27
# ./configure
# make
# make install
# cd /usr/lib
# ln -s /usr/local/lib/sasl .
# ln -s /usr/local/lib/libsasl* .
# echo"pwcheck_method:passwd"/usr/lib/sasl/Sendmail.conf
```

11.2.2 安裝 Sendmail

在安裝 Sendmail 之前，如果系統中正在執行舊版的 Sendmail，請先使用以下指令將它停掉：

```
# kill -9 `cat /var/run/sendmail.pid|head -1`
```

接著請到 www.sendmail.org 去下載最新版的 sendmail，您也可以在光碟二的 /ports/distfiles 目錄中找到Sendmail 8.12.2。Sendmail 自 8.12.0 版起，需要先建立一個使用者smmsp及所屬群組供 Sendmail 使用。而 FreeBSD 自從 4.5-RELEASE 起已經內建了該使用者，如果你使用的是 4.5-RELEASE 以前的版本，請先編輯 /etc/group 加入下面一行：

```
smmsp:*:25:
```

再來增加使用者，執行 vipw 並增加下面一行：

```
smmsp:*:25:25::0:0:Sendmail user:/var/spool/clientmqueue:/sbin/nologin
```

將抓回來的 sendmail.8.12.1.tar.gz 放到 /tmp 底下，並以下列指令解壓縮：

```
# tar zxvf sendmail.8.12.2.tar.gz
```

接著 ee /tmp/sendmail-8.12.2/devtools/Site/site.config.mc 建立檔案並加入下列內容：

```
PREPENDDEF(`confMAPDEF', `-DMAP_REGEX')
PREPENDDEF(`confOPTIMIZE', `-O2')
APPENDDEF(`confENVDEF', `-DTCPWRAPPERS -DSASL')
APPENDDEF(`conf_sendmail_LIBS', `-lwrap -lsasl')
APPENDDEF(`confLIBDIRS', `-L/usr/local/lib')
APPENDDEF(`confINCDIRS', `-I/usr/local/include')
```

完成後就可以開始安裝 Sendmail 了：

```
# cd /tmp/sendmail-8.12.2
# sh Build -c -f /tmp/sendmail-8.12.2/devtools/Site/site.config.mc
# sh Build install
```

爲了要讓本地的使用者不必經由身份認證使用 SMTP，還須再做下列設定：

```
# cd /tmp/sendmail-8.12.2/obj*
# cd mail.local
# make force-install
# chown root /usr/libexec/mail.local
# chmod u+s /usr/libexec/mail.local
```

接下來編輯 sendmail 設定範本檔案以便產生出具有身分認證功能的 sendmail.cf 設定檔，請以下列指令執行：

```
# cd /tmp/sendmail-8.12.2/cf/cf
# cp generic-bsd4.4.mc MYCF.mc
# cd ../feature
# cat access_db.m4 >> ../cf/MYCF.mc
# cat delay_checks.m4 >> ../cf/MYCF.mc
# cat virtusertable.m4 >> ../cf/MYCF.mc
# cd ../cf
# cat >> MYCF.mc
TRUST_AUTH_MECH(`LOGIN PLAIN')dnl
define(`confAUTH_MECHANISMS', `LOGIN PLAIN')dnl
```

```
^D(同時按Ctrl+D)
```

```
# sh Build MYCF.cf
# cp MYCF.cf /etc/mail/sendmail.cf
# cd /etc/mail
# cat > access
127.0.0.1 RELAY
(本機的IP) RELAY
^D (同時按Ctrl+D)
# touch virtusertable
# touch aliases
```

編輯 /etc/mail 下的檔案，新增一個名爲 local-host-names 的檔案，裡面塡入 localhost 以及機器的完整名字 (如 mail.abc.com)，這樣一來由本機發信時便不需再一次做使用者認證。接著，再新增一個檔名爲 relay-domains 的檔案塡入本地的 domainname，例如 mydomain.com，當收到信的收件者不是給 mydomain.com 時便會拒絕。然後將這些檔案做成 sendmail 能接受的 DB 格式：

```
# makemap hash access.db < access
# makemap hash virtusertable.db < virtusertable
# newaliases
```

安裝完後要把以下檔案權限改成這樣：

```
-r-xr-sr-x    root         smmsp ...    /usr/sbin/sendmail
drwxrwx---    smmsp smmsp ...    /var/spool/clientmqueue
drwx------    root         wheel        ...    /var/spool/mqueue
-r--r--r--    root         wheel        ...    /etc/mail/sendmail.cf
-r--r--r--    root         wheel        ...    /etc/mail/submit.cf
```

請使用下列指令來修改上述檔案的權限：

```
# chown smmsp.smmsp /var/spool/clientmqueue
# chmod 770 /var/spool/clientmqueue
# chown root.wheel /var/spool/mqueue
# chmod 700 /var/spool/mqueue
# chown root.wheel /etc/mail/sendmail.cf
# chmod 444 /etc/mail/sendmail.cf
# chown root.wheel /etc/mail/submit.cf
# chmod 444 /etc/mail/submit.cf
```

這樣就完成了。您可以執行 sendmail -d0.1 -bv root | grep SASL 因該會出現 SASL 的字樣，表示已可認証。或者在啓動 Sendmail 之後，執行 telnet localhost 25 ，再打ehlo localhost，我們應該看到一堆 250- 開頭的訊息，其中有一行 250-AUTH LOGIN PLAIN 就代表 Sendmail 已經具有身份認證的功能，此時請輸入 quit 結束。萬一沒有出現，請閱讀 /var/log/maillog 裡面的訊息可以知道錯在哪裡。

最後以 /usr/sbin/sendmail -bd -q30m 來啓動Sendmail。如果我們希望在一開機便啓動 Sendmail，請在 /etc/rc.conf 中加入下面這一行：

```
sendmail_enable="YES"
```



11.2.3 Client 端的設定

微軟的 outlook 由 4.0 版開始支援發信時身分認證功能，只要在 outlook 的設定裡面將『外寄郵件伺服器需要查驗身分』的選項打勾就可以了。

Step1.選取『工具』功能表的『帳號』選項：

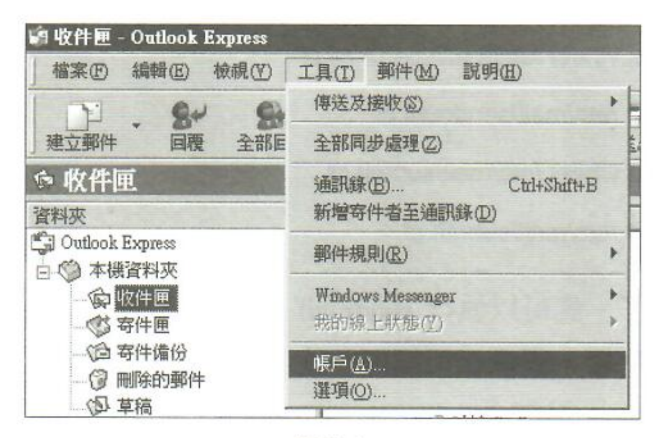


圖 11-1

Step2.選取帳號選單中的『內容』按鈕：



圖 11-2

Step3.將『外寄郵件伺服器需要查驗身分』功能項打勾：

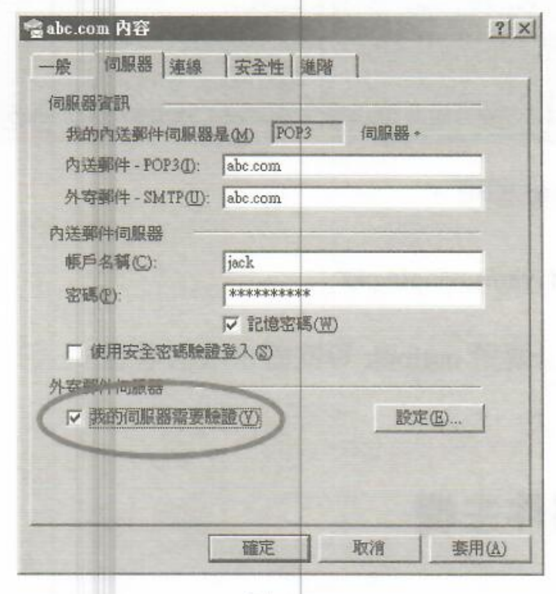


圖 11-3

按確定鈕回到 outlook 中，即可使用身分認證功能發信。

11.3 POP3 設定

POP3 的設定很簡單，只要選一個喜歡的 pop3 軟體，以 ports 安裝完後再做一些設定就好了。在這裡我選用 popa3d：

```
# cd /usr/ports/mail/popa3d
# make install clean
```

接著編輯 /etc/inetd.conf，找到 pop3 的部份，將開頭的 # 拿掉後，並加以修改如下：

```
#
# example entry for the optional pop3 server
#
pop3  stream  tcp  nowait  root  /usr/local/libexec/popa3d   popa3d
```

接著重新跑 inetd 即可：

```
# kill -HUP `cat /var/run/inetd.pid`
```

現在我們就可以使用 outlook 等軟體來收信看看。

11.4 虛擬郵件主機

如果我們想要在同一台機器上收多台主機的信件，或者想要在一台主機上設定可以 "收" "發" 信件的虛擬帳號 (如果只要收，可以簡單的設定 aliases 即可)，我們可以經由虛擬主機的設定來達成。假設有二個 Domain Name，一個是 abc.com，另一個是 old.cde.com。這份文件包含了二種設定方式：一個是讓二個 domain name 收到的信對映到一台機器上的使用者，也就是說不能有不存在的虛擬使用者；另一個設定是讓你可以設定不同的虛擬使用者對映到不同機器上的任何使用者。不管我們要做哪一種設定，都必需要先設定 DNS。

請注意，這份文件中的設定並不會讓你可以擁有一個眞正的虛擬帳號，因爲這裡是將虛擬帳號對映到一個存在的郵件位址。使用這份文件的設定和設定 /etc/aliases 最大的不同在於設定 aliases 只能讓虛擬的帳號收信，而無法寄信。

11.4.1 DNS 設定

爲了要讓寄出去的信知道要往哪一台主機上送，必須要先設定 DNS。假設我們現在已經有一台設定好 DNS 的主機，hostname 是 abc.com。我們要讓 abc.com 處理 old.cde.com 的信件的話，最簡單的方式就是將 mail.cde.com 指向 abc.com (CNAME records)，也就是二個 doamin 有同樣的 IP。不過這樣一來，old.cde.com 就不能獨立存在了，也就是說不會有一台機器的 hostname 名爲 old.cde.com，並提供 FTP、www(也可以有虛擬主機)、telnet等服務。

因此，我們要使用的是改變 DNS 的 MX record。設定只有處理該主機的郵件時，才將 old.cde.com 轉向 abc.com。請在你的 DNS 中加入下列設定：

```
old.cde.com        IN  MX  10   abc.com.
```

完成後要重新讀設定檔並等一段時間設定才會在網路上生效。接著我們就可以來做 sendmail 的設定了。

11.4.2 對映到同一台機器的真實使用者

第一種設定的使用時機，例如你的公司主機原來是 mail.cde.com，現在換成了 abc.com，你希望讓原本的使用者 jack@mail.cde.com 和新的 jack@abc.com 都能由 jack@abc.com 來收信。這種設定很簡單，只要編輯 /etc/mail/relay-domains 及 /etc/mail/local-host-names 這二個檔案，加入要增加收信的主機名稱即可。以本例而言，除了那二個檔案原本的內容外，要再增加一行：

```
mail.cde.com
```

這樣了不管是原本 mail.cde.com 或是眞正主機名稱 abc.com 的信件，都可以由 abc.com 的相同的使用者收信。

11.4.3 可以擁有虛擬使用者

第二種設定是讓我們可以設定一個虛擬的帳號，並且可以利用它來送信。這個設定是經由編輯 /etc/mail/virtusertable 來達成。相同的，我們在 abc.com 這台主機中設定它的 virtusertable。我們以下列二個 virtusertable 的例子來說明，請注意，二個欄位間的空白是用 tab 鍵，而非使用空白鍵：

範例一：

```
joe@mail.cde.com	jschmoe
jane@mail.cde.com	jdoe@othercompany.com
@mail.cde.com	jschmoe
```

上面的例子中，凡是寄給 joe@mail.cde.com 的信都會送給本地使用者 jschmoe。而以 joe@mail.cde.com 寄出的信收件人所看到的寄件人一樣是 joe@mail.cde.com，如果在寄信時要身份認證的話，必須使用 jschmoe 的帳號及其密碼。接下來，寄給 jane@mail.cde.com 的信會送給 jdoe@othercompany.com，剩下來所有給 mail.cde.com 的信都會送給本地 jschmoe 這個使用者。

範例二：

```
joe@mail.cde.com	jschmoe
bogus@mail.cde.com	error:nouser No such user here
list@mail.cde.com	yourdomain-list
@mail.cde.com	%1@othercompany.com
```

這一個例子中，第一行和範例一一樣，凡是寄給 joe@mail.cde.com 的信都會送給本地使用者 jschmoe。而凡是寄給 bogus@mail.cde.com 都會回

應沒有這個使用者。第三行如果是寄給 list@mail.cde.com 的信，都會轉給本地的 yourdomain-list 這個虛擬使用者，你可以在 /etc/aliases 中加入關於 yourdomain-list 這個使用者的信要怎麼處理，怎麼設定別名。最後一行，凡是其他非上述三行使用者的信，都交由在 othercompany.com 這台機器上相對映的使用者來處理。

你可以依照上面的範例來編輯你的 virtusertable，完成編輯後，必須要使用以下指令來將這些檔案做成 sendmail 能接受的 DB 格式：

```
# makemap hash virtusertable.db < virtusertable
# newaliases
```

都完成後，我們就可以到別台機器使用虛擬帳號來試試收發信。假設我們要使用的虛擬帳號是上述範例一中的第二行 jane@mail.cde.com，以 outlook 中的設定爲例，所設定的 E-mail 仍然是 jane@mail.cde.com，如圖 11-4所示：

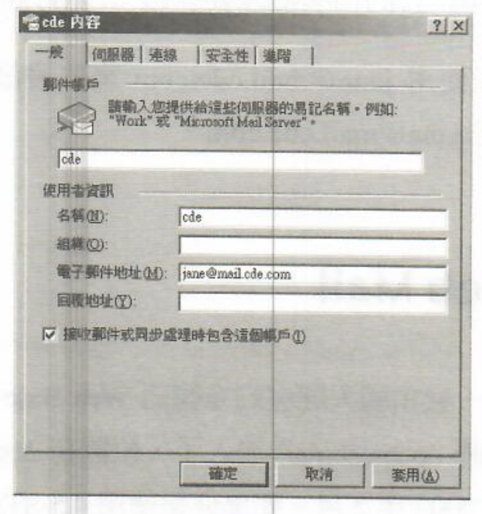


圖 11-4

所設定的 pop3 及 smtp 主機也是 mail.cde.com。但是使用者及密碼是

othercompany.com 上的使用者 jdoe 及其密碼，如圖11-5所示：

圖 11-5

在上圖中，如果 othercompany.com 在寄信時要身份認証，則上圖中 "外寄郵件伺服器" 的選項 "我的伺服器需要驗證"必須打勾。

如此一來您就可以使用 jane@mail.cde.com 來收發信，而且在別人收到信時會顯示寄件人是 jane@mail.cde.com。

11.5 Open Web Mail

Open Web Mail 是一套由國人開發的多國語 Web Base 的郵件軟體。 現在最新的 ports 中已經將 opebwebmail 加入了，位置在 /usr/ports/mail/openwebmail，加入的日期是 2002 年 2 月。如果你是使用 4.5-STABLE 以後的版本，就可以直接用 ports 來安裝了。不過 ports 預設的網頁目錄位於 /usr/local/htdocs 中，可能不符合我們的需求，因此我們還是使用自己編譯。

11.5.1 系統需求

你必須先安裝具有 CGI 功能的 Apache 伺服器，如果你是照本書的說明安裝，那麼你的伺服器就具有這項功能了，只要你有編輯 httpd.conf 將 CGI 的功能啓動。

爲了要有檢查附加檔案的功能，還要安裝 MIME-Base64-2.12.tar.gz，你可以在本書光碟二的 /ports/distfiles 目錄中取得這裡所需的所有檔案。將檔案放到 /tmp 後，使用下列指令安裝：

```
# tar zxvf MIME-Base64-2.12.tar.gz
# cd MIME-Base64-2.12
# perl Makefile.PL
# make
# make test
# make install
```

爲了具有拼字檢查功能，必須安裝 ispell，請以 ports 安裝：

```
# cd /usr/ports/textproc/ispell
# make install
```

另外，還要先安裝 libnet 這個模組：

```
# tar -zxvf libnet-1.0901.tar.gz
# cd libnet-1.0901
# perl Makefile.PL
# make
```

```
# make install
```

現在已經做好事前的準備了。

11.5.2 安裝 Open Web Mail

您可以使用本書所附的 1.62 版或是到下列網址取得最新版的 Open Web Mail。

http://turtle.ee.ncku.edu.tw/openwebmail/download/

假設網頁根目錄在 /home/www 中，而在 apache 中所設定的 cgi-bin 目錄是在 /home/www/cgi-bin 中。將取得的檔案放到你的網頁根目錄去，並 cd 到你的網頁根目錄。執行下列指令以將取得的檔案解壓縮：

```
# tar -zxvBpf openwebmail-1.62.tgz
```

解壓縮後會在 cgi-bin 中產生一個目錄爲 openwebmail，存放 openwebmail 的主要程式；另外會產生一個 data 的目錄，在 data 目錄下也有一個 openwebmail 的目錄，該目錄存放 openwebmail 非 cgi 的資料(如圖片、聲音等)。我將 /home/www/data/openwebmail 的目錄搬到/home/www/openwebmail。現在 openwebmail 的 cgi 程式位於 /home/www/cgi-bin/openwebmail 中，非 cgi 檔案位於 /home/www/openwebmail 中，我們要記住的就是這二個目錄的位置。接下來修改權限：

```
# chown -R root.mail /home/www/cgi-bin/openwebmail/
# chmod 750 /home/www/cgi-bin/openwebmail/etc
# chmod 770 /home/www/cgi-bin/openwebmail/etc/sessions
```

```
# chmod 770 /home/www/cgi-bin/openwebmail/etc/users
# chmod 4755 /usr/bin/suidperl
```

最後要修改 /home/www/cgi-bin/openwebmail/etc/openwebmail.conf，更改你的參數設定，尤其是路徑。 基本上要修改的地方有：

```
# ow_cgidir : openwebmail cgi 程式的目錄
ow_cgidir /home/www/cgi-bin/openwebmail
# ow_cgiurl : 以 openwebmail 的 cgi 程式目錄所在的 url
ow_cgiurl /cgi-bin/openwebmail
# ow_htmldir : openwebmail 非 cgi 的目錄
ow_htmldir /home/www/openwebmail
# ow_htmlurl : 非 cgi 程式所在的 url
ow_htmlurl /openwebmail
```

修改完後就可以使用 http://yourhost/cgi-bin/openwebmail/openwebmail.pl 來連到登入的首頁，請使用系統中的使用者帳號及密碼登入，登入後畫面如圖11-6所示。

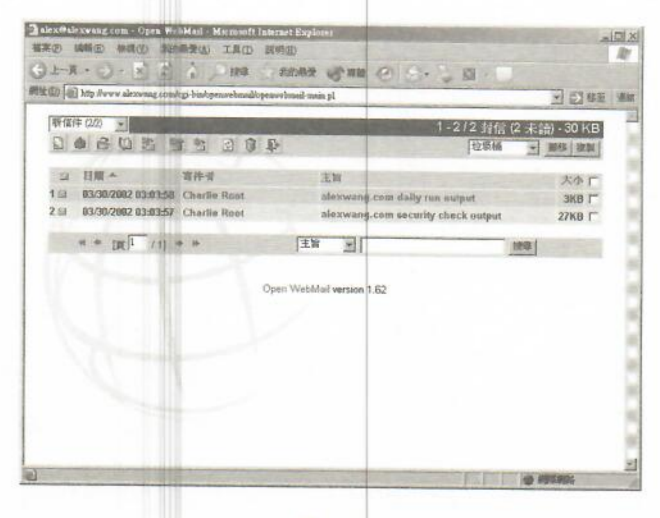


圖 11-6

chapter 12

DNS伺服器

12.1 DNS 概論

DNS (Domain Name Service) 是網域名稱服務的縮寫，其主要目地是在解決機器的網域名稱 (Domain name) 與 IP address 的對應問題。 在網際網路上，為了要連線到其他電腦，必須經由 IP 位址來判斷電腦所在位置，例如 140.115.83.240 就是一個 IP 位址。但是這一長串的 IP 並不好記，因此出現了 Domain Name 來為 IP 取一個比較好記的名字，如 bbs.mgt.ncu.edu.tw。 Domain Name 的架構是一個樹狀結構，例如上述的 bbs.mgt.ncu.edu.tw 所代表的就是台灣 (tw) 的中央大學 (ncu) 管理學院 (mgt) 所屬的電子佈告欄伺服器 (bbs)。而 DNS 伺服器的功用就是將你輸入的 Domain Name 轉成 IP，或者是將查詢 IP 並轉回所對映的 Domain Name。

一般而言， DNS 伺服器可以分為三種，主要名稱伺服器 (Primary/master Server)，次要名稱伺服器 (Secondary/slave Server)，及快取名稱伺服器 (Cache only Server)。主要名稱伺服器是管理所屬網域所需名稱對映設定的主要伺服器，如果您自己有一個網域，必須經由設定主要名稱伺服器來管理網域中 IP 所對映的名稱。而次要名稱伺服器是取得主要名稱伺服器的資料，用以在主要伺服器過於忙錄或停止服務時備用。每個 DNS 伺服器都會將所查詢過的 Domain Name 建立快取 (cache)，以供下次查詢時能快速回應。每一個伺服器都會設定該 Domain Name 快取的資料要保留多久，以免得到過時的資料。

DNS 有分為正解 (forward) 及反解 (reverse)。正解就是把 Domain Name 轉成 IP，而反解是將 IP 轉成 Domain Name。 FreeBSD 內建有 DNS 的服務，只要先設定 /etc/namedb/ 下的檔案即可打開該服務。

那要如何得到一個 Domain Name 呢？以學校而言，每個學校都有自己的 Domain Name 及 IP 範圍，如果你只有一台機器，想要爲它申請一個 Domain Name，您並不需要自己架設 DNS。只要向學校計算機中心或是管理該 IP 的單位提出申請正反解即可。如果您想爲自己的系所架設 DNS 以管理該系的 IP，必須先向上層 DNS (也就是校方計中) 申請授權，讓針對您系所的查詢都交由你的 DNS 伺服器處理。DNS 的架構是樹狀結構，以查詢 bbs.mgt.ncu.edu.tw 而言，管理 tw 網域的伺服器會告訴你管理 ncu.tw 的伺服器在哪裡，而 ncu.tw 會告訴你管理 mgt.ncu.edu.tw 的伺服器在哪裡，最後 mgt.ncu.edu.tw 發現自己有 bbs.mgt 的資料，並傳回其 IP。

如果您自己要申請一個 Domain Name，你可以到 www.twnic.com.tw 去申請一個 .tw 的名稱，如 abc.com.tw；或者到 www.register.com (國外一個不錯的申請網站) 的去申請各國的 Domain，如 abc.com。當您去 TWNIC 申請網域名稱時，你必須要有自己的 DNS 伺服器，並在 TWNIC 設定 DNS 伺服器的位址，接著再由您的 DNS 伺服器來做解析。如果您是在 www.register.com 申請網域名稱，國外的代理申請者會幫你做 DNS 的服務，你只要在他們的網頁設定你想要的名稱及所對映的 IP 即可，而且不限個數，當然你也可以自行架設 DNS 伺服器。

如果只有一個固定 IP，您不需要架設 DNS，除非我們管理一個網域中多個 IP 才有必要。再者，如果您是固接式 ADSL 或是其他由 ISP 提供的連線方式，你的 DNS 反解必須要由上層 ISP 授權由您自行管理你的 IP ，這樣你的設定才會生效，否則應該請 ISP 幫你做反解的設定，也就是告訴 ISP 你的 IP 要對映那一個 Domain Name。

對於 DNS 有初步概念之後，我們接著說明如何設定 FreeBSD 成爲一台 DNS 伺服器。我們以下列的資料做爲設定的依據：

網域：abc.com

IP：123.44.55.224 ~ 123.44.55.231 (八個 IP)

Netmask：255.255.255.248

首先，到 /etc/namedb 的目錄下：

```
# cd /etc/namedb
# sh ./make_localhost
```

這個指令將產生一個名為 localhost.rev 的檔案，該檔案是用來反解 localhost 的設定檔。接著請編輯 named.conf 來定義我們要設定的資料。

12.2 named.conf

我們先在原本 /etc/named.conf 文件的最下方加入下列設定，該檔中行首以 // 開頭者為註解：

```
// 設定要正解的網域是 abc.com，DNS 伺服器是 master server
// 並到 abc.fwd 這一個檔案中去找 abc.com 的設定。
zone "abc.com" {
  type master;
  file "abc.fwd";
};

// 下面是反解的資料，設定要反解的 IP 為 123.44.55.224 到 123.44.55.231
// 設定關於該 IP 範圍反解的設定檔為 abc.rev
zone "224-29.55.44.123.in-addr.arpa" {
```




```
    type master;
    file "abc.rev";
};
```

接下來就可以新增並編輯 abc.fwd 這個正解檔及 abc.rev 反解檔。

如果你的 IP 的範圍是整個 Class C，子網路遮罩 (Netmask) 為 255.255.255.0，或是其他經由切割過的 IP，關於上面反解資料 zone 那一行的設定會有點不同。

以 123.44.55.0 為例，假設我們具有完整 Class C ，也就是 123.44.55.*：

```
zone "55.44.123.in-addr.arpa"
NetMask:255.255.255.0

若是 Class C 分成2份，每一段有128個IP
zone  "0-25.55.44.123.in-addr.arpa" ; 0-25 是第 1段
zone "128-25.55.44.123.in-addr.arpa" ;128-25 是第 2段
NetMask:255.255.255.128

若是 Class C 分成4份，每一段有64個IP
zone  "0-26.55.44.123.in-addr.arpa" ; 0-26 是第 1段
zone  "64-26.55.44.123.in-addr.arpa" ; 64-26 是第 2段
zone "128-26.55.44.123.in-addr.arpa" ;128-26 是第 3段
zone "192-26.55.44.123.in-addr.arpa" ;192-26 是第 4段
NetMask:255.255.255.192
```

12.3 正解檔設定

我們可以複製 localhost.rev 來加以修改，以下是 abc.fwd 的內容，檔案中 ";" 之後為註解：

```
$TTL  172800

@     IN   SOA     abc.com. root.abc.com. (
                          2001080301 ; Serial
                          172800 ; Refresh
                          900       ; Retry
                          3600000          ; Expire
                          3600 )  ; Minimum
      IN   NS      abc.com.
      IN   A       123.44.55.225
www   IN   A       123.44.55.226
ftp   IN   CNAME   www
mail  IN   A       123.44.55.227
mail  IN   MX      10 www.abc.com
      IN   MX      20 mail.def.net
```

接著我們針對檔案中的每一行加以解釋，您可以依自己的需要來增刪資料。這個檔案的內容格式為 [name] [ttl] [class] [type] [data]。

[name] 可以是網域名稱或是主機名稱，如果不寫的話表示與上一個設定相同。

[ttl] 是資料要存活的時間 (time to live)，也就是 cache server 將保留在它的 cache 中的時間。如果不寫的話表示和 SOA 中的設定相同。

[class] 指定網路的類型，應該使用 IN 代表 internet。
[type] 設定該筆資料的型態，例如：MX, A, CNAME, PTR, NS 等。
[data] 代表這設定的資料要存多久，以秒數計算。172800 為二天。

@ IN SOA abc.com. root.abc.com. ( ) 那一行中，@ 代表網域名稱 abc.com，IN 表示為 internet 的資料型態。SOA 後面接的是 abc.com，表示這台 abc.com 機器是 abc.com 網域中的主要名稱伺服器。而 root.abc.com 表示管理者的Email 是 root@abc.com。

Serial：這個設定的版本，這次修改的數字必須比上次的數字大，也就是每次修改這個檔時，都要將這個數字提高，這樣別的伺服器才會將資料更新。一般而言，我們會以日期加上幾位的數字來表示，如 2001082101。

Refresh：這個數字是次要名稱伺服器要多久和主要名稱伺服器比對資料並更新。

Retry：如果比對失敗，要在幾秒後再向主要名稱伺服器查詢。

Expire：表示如果次要名稱伺服器一直連不上主要名稱伺服器，這筆資料要多久無法比對便失效。一樣是以秒計。

Minimum：表示別的快取伺服器可以將你的設定存放多久。

接下來的 IN NS abc.com. 表示將 abc.com 這個網域的 DNS 伺服器是 abc.com 這台機器。

IN A 123.44.55.225 表示將 abc.com 這台機器的 IP 設為 123.44.55.225。前面省略了主機名稱，表示設定的是 @ 的主機。A 代表的就是指定 address。



www IN A 123.44.55.226 表示將 www.abc.com 的 IP 設定為 123.44.55.226。

接著是 CNAME，表示將 ftp.abc.com 這台機器設定為和 www.abc.com 同一個 IP。也就是這二個名字會使用同一台機器。

MX 表示將 mail.abc.com 的信件交由 www.abc.com 或 mail.def.net 來處理，數字 10 及 20 表示優先順序，數字小者優先。這裡的設定是將給 mail.abc.com 的信件交給 www.abc.com 來處理，如果 www.abc.com 沒有回應則交由 mail.def.net 來處理。

12.4 反解檔設定

如果您要設定 DNS 反解，則需要再新增並編輯 abc.rev 的檔案來設定，也就是設定某一個 Domain Name 要對應到哪個 IP：

```
$TTL  172800
@     IN    SOA   abc.com. root.abc.com. (
                        2001080301  ; Serial
                        172800 ; Refresh
                        900     ; Retry
                        3600000         ; Expire
                        3600 )  ; Minimum
      IN    NS    abc.com.
224   IN    PTR   UNKNOW.abc.com.
225   IN    PTR   abc.com.
226   IN    PTR   www.abc.com.
227   IN    PTR   mail.abc.com.
```

```
228    IN    PTR    UNKNOW.def.net.
226    IN    PTR    UNKNOW.abc.com.
227    IN    PTR    UNKNOW.abc.com.
228    IN    PTR    UNKNOW.def.net.
```

和正解檔重覆的地方此略過，我們來看 PTR 的部份。PTR 就是將 IP 指向 Domain Name，如 225 IN PTR abc.com. 就是將 123.44.55.225 指向 abc.com 這台機器。

12.5 最後的設定

正解檔和反解檔都修改好之後，請先編輯 /etc/resolv.conf，在 nameserver 部份的第一行加入：

```
nameserver 127.0.0.1
```

以 /usr/sbin/named 這個指令來啓動 DNS 的服務。如果有錯誤，請以

```
# tail /var/log/messages
```

來看錯誤訊息。如果不是第一次執行 named 就以 kill -1 來將它重跑。接著就可以用 nslookup 來試試看你的設定。

```
# nslookup www.abc.com
# nslookup 123.44.55.225
```

如果你沒有設定 /etc/resolv.conf 的話，上面的指令並不會以你的機器爲預設 DNS 伺服器，這時你可以使用下列指令來以 abc.com 這台機器爲伺服器，查詢 www.abc.com 的設定：

nslookup www.abc.com abc.com

如果您要查詢 MX、NS、SOA 等記錄，也可以使用 nslookup 來查詢，只要在查詢先 set type 即可：

```
# nslookup
Default Server:  localhost.abc.com
Address:  127.0.0.1
> set type=MX
> mail.abc.com
Server:  localhost.abc.com
Address:  127.0.0.1
mail.abc.com        preference = 10, mail exchanger = www.abc.com
abc.com     nameserver = e368.com
www.abc.com internet address = 123.44.55.225
abc.com     internet address = 123.44.55.225
> set type=NS
> abc.com
Server:  localhost.abc.com
Address:  127.0.0.1

abc.com       nameserver = abc.com
abc.com       internet address = 123.44.55.225
> set type=SOA
> abc.com
Server:  localhost.abc.com
Address:  127.0.0.1
```

```
abc.com
      origin =abc.com
      mail addr = root.abc.com
      serial = 2002080301
      refresh = 3600 (1H)
      retry   = 900 (15M)
      expire  = 3600000 (5w6d16h)
      minimum ttl = 3600 (1H)
abc.com        nameserver = abc.com
abc.com        internet address = 123.44.55.2.225
> exit
```

所有設定都沒有問題之後，如果要在開機時就啟動 DNS 服務，請在 /etc/rc.conf 中加入：

```
named_enable="YES"
```

chapter 13

NAT及防火牆



13.1 概論

這個部份我們將說明如何以 FreeBSD 做為防火牆，介紹 FreeBSD 內建的封包過濾功能。欲建立一台防火牆，就是要將一台機器放在二個網域的中間，並經由它來做封包過濾的工作。 因此我們必須先確定網路封包能通過這台防火牆，再來設定要阻檔的規則。以 FreeBSD 作為二個網域中間的連接器，可以用路由器 (router)、閘道、或是橋接器的方式來實作，再在該機器上設定防火牆的規則。當然，我們也可以只在一台單機上設定防火牆規則，以取代原本只能監控 inetd 服務的 TCP Wrapper。

我們舉二種最常被應用結合防火牆設定來保護整個網路的方法，一個是 NAT，另一個是具封包過濾的橋接器。

13.1.1 NAT

所謂的 NAT 就是 (Network Address Translation)，它可以讓你在只有一個 IP 的情形下讓多台電腦一起連上網路。舉個實例而言，一個公司有三十台電腦，卻只有八個 IP，可以將每台電腦的 IP 設定為 private IP，再讓它們經由一台有 IP 的 NAT 伺服器連上網路即可。

Private IP 是 RFC 所定義的私人 IP，這些 IP 不能夠直接在網際網路中出現，所以必需經由 NAT 的轉換，將它們偽裝成是由 NAT 伺服器連向外部網路。這些可以用的私人 IP 如下：

Class A	10.0.0.0	- 10.255.255.255	255.0.0.0
Class B	172.16.0.0	- 172.31.255.255	255.255.0.0
Class C	192.168.0.0	- 192.168.255.255	255.255.255.255

我們只需在 NAT 伺服器中做好設定，再將其他使用 private IP 的電腦設定 gateway 為該伺服器的 IP 即可。另外，我們也可以在伺服器中設定一些防火牆的規則，來保全內部網路。

13.1.2 具封包過濾的橋接器

如果我們的網路中有多台不同網域的電腦，這些電腦都有它們的 IP 及網路設定，我們可以將 FreeBSD 設定成為橋接器 (bridge)，讓這台橋接器作封包過濾的工作。這種做法對於網域內其他電腦原本的網路設定不會有影響，如果沒有設定任何防火牆規則，對它們而言幾乎不會發現橋接器的存在。

我們也可以使用路由器來取代橋接器，但是路由器只能遶送二個不同網域，而且設定比較複雜，因此，我會使用橋接器來做為防火牆。

FreeBSD 內建有 ipfw 這個程式可以讓我們輕易的設定一個簡單的防火牆，我們只要在 kernel 中加上一些設定就可以打開它。在這裡我們也將簡單的介紹一些防火牆的語法，讓我們可以保護我們不想、不需要被外界使用的網路服務。

在設定防火牆之前，有個觀念必須先釐清。防火牆並不能夠完全保護我們的網路安全，防火牆只是限制我們不想公開的服務、限制已知的 IP。就算架了防火牆，沒有適當的管理也是枉然。

13.2 NAT

這裡我們假設使用二張網路卡，一張是對外的網卡，代號是 vr0；另一張是對內的網卡，代號 vr1。以下的設定中請依您的網卡代號來加以修改。當然，你也可以只使用一張網路卡，將所有的電腦及對外網路都接在一台 HUB 上，再利用 alias 的功能將一張網卡設定二個 IP。

13.2.1 設定 kernel

請先在 kernel 中加入下列幾行，並重新編譯 kernel，如果您不知道如何修改 kernel，請參考第三章「如何編譯核心」。 假設要修改的 kernel 是 /usr/sys/i386/conf/GENERIC，先 cd /usr/sys/i386/conf/，再 ee GENERIC 加入下列幾行：

```
# 防火牆
options         IPFIREWALL
# 支援 nat
options         IPDIVERT
# 下面這一行是預設允許所有封包通過，如果沒有這一行，就必須在
# /etc/rc.firewall 中設定封包的規則。
options         IPFIREWALL_DEFAULT_TO_ACCEPT
# 下面這一行是讓你可以在 ipfw 中設定要記錄哪些封包，
# 如果沒有這一行，就算設定了要留下記錄也不會有作用。
options         IPFIREWALL_VERBOSE
```

存檔後，執行下列指令以安裝新的 kernel：

- # config GENERIC
- # cd ../../compile/GENERIC/
- # make depend
- # make
- # make install

這樣子我們的核心就已經支援防火牆及 NAT 了。

13.2.2 設定 rc.conf

接著要修改 /etc/rc.conf 加入下列幾行，我們假設網路卡代號是 vr0 及 vr1，請自行變更成您的網路卡代號：

```
# 對外網路的設定，假設 IP 是 123.44.55.66
ifconfig_vr0="inet 123.44.55.66 netmask 255.255.255.0"
# NAT 的設定
ifconfig_vr1="inet 192.168.0.1  netmask 255.255.255.0"
# 如果只使用一張網路卡的話，改用下面這一行
#ifconfig_vr0_alias0="inet 192.168.0.1 netmask 255.255.255.0"

gateway_enable="YES"
firewall_enable="YES"
firewall_type="OPEN"
natd_interface="vr0"
natd_enable="YES"
```

設定結束之後，重開機應該就可以設定其他電腦使用這台 NAT 伺服器來連上網路了。如果不行，請先檢查 /etc/service 中是不是有下面這一行：

```
natd   8668/divert
```

還是不行的話，就是還要再修改 /etc/rc.firewall。我們在 /etc/rc.conf 中設定 firewall_type="OPEN"，如果是使用原本 /etc/rc.firewall 的話，這樣就已經就已經驅動了 NAT 的功能，不然的話，請再看下列 firewall 的設定。

13.2.3 設定 rc.firewall

爲了使 NAT 能運作，必須編輯 /etc/rc.firewall ，另外我們也可以在 rc.firewall 中加入一些防火牆的設定，來阻擋一些機器或服務。先備份原本的 rc.firewall，再修改其內容如下：

```
#!/bin/sh
# 清除所有的規則
/sbin/ipfw -f flush
# 在此加入拒絕的規則
/sbin/ipfw deny log all from 61.55.121.1 to any
/sbin/ipfw deny log all from any to 61.55.121.1
# NAT
/sbin/ipfw add divert natd all from any to any via vr0
# 允許其他所有封包通過
/sbin/ipfw add pass all from any to any
```

修改完後，執行 sh /etc/rc.firewall 就可以更新防火牆的設定了。其他關於防火牆規則的詳細說明，請 man ipfw 或參考下一節的說明。

13.2.4 client 端的設定

接者是 client 端方面，除了上述的 NAT 設定外，在其他電腦還要再做一些設定。首先，你的網路架構應該如圖13-1所示：

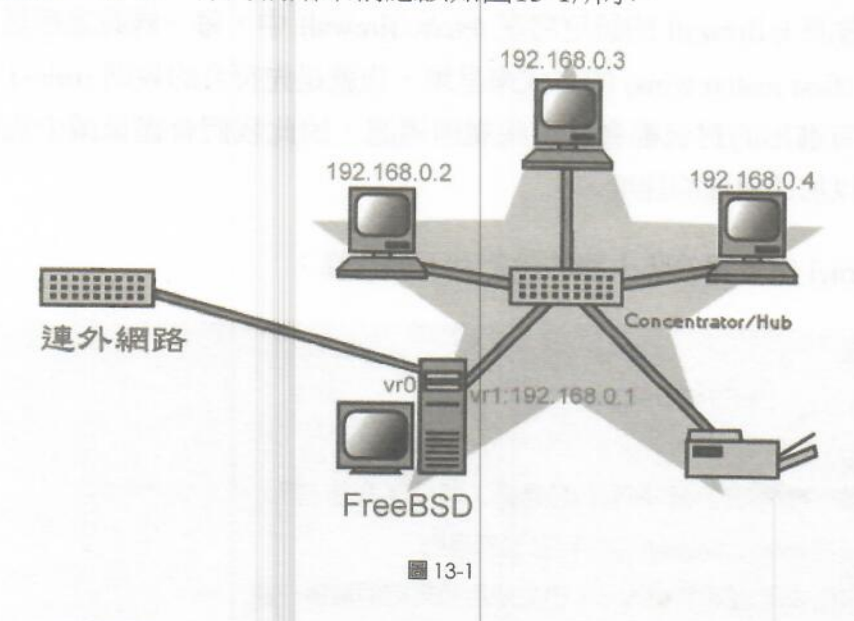


圖 13-1

其他的電腦設定方面，我們必須將其他電腦的 IP 設定爲 192.168.0.X、子網路遮罩是 255.255.255.0，gateway 設定爲 FreeBSD 連到區域網路的網路卡 IP，在此範例中是 192.168.0.1。然後再設定你的 DNS 爲你 ISP 的 DNS。

完成上述的設定後，我們就能享受以 FreeBSD 爲 NAT上網了。

13.3 防火牆

ipfw 是 FreeBSD 內建的防火牆指令，我們可以用它來管理進出的網路交通。如果防火牆伺服器是扮演著路由器 (gateway 例如上一篇中的 NAT 伺服器) 的角色，，則進出的封包會被 ipfw 處理二次，而如果防火牆扮演的是橋接器 (bridge) 的角色，則封包只會被處理一次。這個觀念關係著我們以下所要介紹的語法，有的語法並不適用於橋接器。

我們會將 ipfirewall 的設定寫在 /etc/rc.firewall 中，每一條設定都是以先入為主 (first match wins) 的方式來呈現，也就是先符合的規則 (rules) 為優先。所有進出的封包都會被這些規則過濾，因此我們會盡量減少規則的數量，以加速處理的速度。

在 kernel 中，關於防火牆的設定有下列幾條：

```
# 防火牆
options         IPFIREWALL

# 下面這一行是預設允許所有封包通過，如果沒有這一行，
# 就必須在 /etc/rc.firewall 中設定封包的規則。
# 這條規則內定編號是 65535，也就是所有規則的最後一條
# 如果沒有加這一條規則，內定就是拒絕所有封包，
# 只允許規則中允許的封包通過。
options         IPFIREWALL_DEFAULT_TO_ACCEPT

# 這一行是讓你可以在 ipfw 中設定要記錄哪些封包，
# 如果沒有這一行，就算設定了要留下記錄也不會有作用。
options         IPFIREWALL_VERBOSE
```

```
# 這一行是限制每一條規則所要記錄的封包數量，
# 因為同樣的規則可能有許多記錄，加上這一條可以使
# 同樣的記錄重覆數減少，以避免記錄檔爆增。
options IPFIREWALL_VERBOSE_LIMIT=10

# 下面這一行是用來支援封包轉向，
# 當你要使用 fwd 動作時必須要有這一項設定。
#options IPFIREWALL_FORWARD

# 如果要使用 pipe 來限制頻寬，必須加入下列選項以支援 dummynet。
options DUMMYNET
```

ipfw 也支援狀態維持 (keep-state) 的功能，就是可以讓符合設定的規則以動態的方式來分配增加規則 (位址或連接埠) 來讓封包通過。也就是說防火牆可以記住一個外流的封包所使用的位址及連接埠，並在接下來的幾分鐘內允許外界回應。這種動態分配的規則有時間的限制，一段時間內會檢查連線狀態，並清除記錄。

所有的規則都有計數器記錄封包的數量、位元數、記錄的數量及時間等。而這些記錄可以用 ipfw 指令來顯示或清除。

在說明 ipfw 規則的語法之前，我們先來看這個指令的用法。ipfw 可以使用參數：

表18

ipfw add [rule]	新增一條規則。
ipfw delete [number]	刪除一條編號為 number 的規則。
ipfw -f flush	清除所有的規則。
ipfw zero	將計數統計歸零。
ipfw list	列出現在所有規則，可以配合下列參數使用。

-a	使用 list 時，可以列出封包統計的數目。
-f	不要提出確認的詢問。
-q	當新增 (add)、歸零(zero)、或清除 (flush) 時，不要列出任何回應。當使用遠端登入，以 script (如 sh /etc/rc.firewall) 來修改防火牆規則時，內定會列出你修改的規則。但是當下了 flush之後，會立即關掉所有連線，這時候回應的訊息無法傳達終端機，而規則也將不被繼續執行。此時唯一的方法就是回到該電腦前重新執行了。在修改防火牆規則時，最好在電腦前修改，以免因為一個小錯誤而使網路連線中斷。
-t	當使用 list 時，列出最後一個符合的時間。
-N	在輸出時嘗試解析 IP 位址及服務的名稱。
-s [field]	當列出規則時，依哪一個計數器 (封包的數量、位元數、記錄的數量及時間) 來排序。

13.3.1 ipfw 規則

我們在過濾封包時，可以依據下列的幾個封包所包含的資訊來處理該封包：

- **接收或傳送的介面，可以使用介面名稱或位址**
- **方向，流入或流出**
- **來源或目的地的 IP 位址，也可以加上子網路遮罩**
- **通訊協定，TCP,UDP,ICMP 等**
- TCP flags
- IP fragment flag
- IP options
- **ICMP 的類型**
- **和封包相關的** socket User/group ID

使用 IP 位址或 TCP/UDP 的埠號來做爲規則可能蠻危險的，因爲這二種都有可能被以假的資訊所蒙騙 (spoof)。但是這二種卻也是最常被使用的方法。

下列爲 ipfw rules 的語法：

```
[number] action [log] proto from src to dist [interface_spec] [option]
```

使用 [ ] 包起來的表示可有可無，我們一一爲大家說明它們的意義：

number：定義規則的順序，因爲規則是以先入爲主的方式處理，如果你將規則設定放在一個檔案中 (如 /etc/rc.firewall)，規則會依每一行排列的順序自動分配編號。你也可以在規則中加上編號，這樣就不需要按順序排列了。如果是在命令列中下 ipfw 指令來新增規則的話，也要指定編號，這樣才能讓規則依我們的喜好排列，否則就會以指令的先後順序來排。這個編號不要重覆，否則結果可能不是你想要的樣子。

action： 可以用的 action 有下列幾個：

表19

allow	允許的規則，符合則通過。也可以使用 pass,permit, accept 等別名。
deny	拒絕通過的規則。
reject	拒絕通過的規則，符合規則的封包將被丟棄並傳回一個 host unreachable 的 ICMP。
count	更新所有符合規則的計數器。
check-state	檢查封包是否符合動態規則，如果符合則停止比對。若沒有 check-state 這條規則，動態規則將被第一個 keep-state 的規則所檢查。
divert port	將符合 divert sock 的封包轉向到指定的 port。
fwd ipaddr	將符合規則封包的去向轉向到 ipaddr，ipaddr 可以是 IP 位址或是 [,port]hostname。如果設定的 ipaddr 不是直接可以到達的位址，則會依本機即有的 routing table 來將封包送出。如果該位址是本地位址(local address)，則保留本地位址並將封包送原本指定的 IP 位址。這項設定通常用來和 transparent proxy 搭配使用。例如：

	ipfw add 50000 fwd 127.0.0.1,3128 tcp from 192.168.1.0/24 to any 80 如果沒有設定 port ，則會依來源封包的 port 將封包送到指定的IP。 使用這項規則時，必須在 kernel 中設定選項 IPFIREWALL_FORWARD。
pipe pipe_nr	傳遞封包給 dummynet(4) "pipe"，用以限制頻寬。使用本語法必須先在核心中加入 option DUMMYNET。請 man ipfw 及 man dummynet。 基本語法是先將要設定頻寬的規則加入： ipfw add pipe pipe_nr .... 再設定該規則的頻寬： ipfw pipe pipe_nr config bw B delay D queue Q plr P 這裡的 pipe_nr 指的是 pipe 規則編號，從 1~65535；B 是指頻寬，可以表示為 bit/s、Kbit/s、Mbit/s、Bytes/s、KBytes/s、或 MBytes/s。D 是延遲多少 milliseconds (1/1000)。Q 是 queue size 的大小 (單位為 packages 或 Bytes)。P 是要隨機丟棄的封包數量。 例如我們要限制內部網域的電腦對外上傳的最大頻_是 20KBytes： ipfw add pipe 1 ip from 192.168.0.1/24 to any in ipfw pipe 1 config bw 20KBytes/s

log：如果該規則有加上 log 這個關鍵字，則會將符合規則的封包記錄在 /var/log/security 中。前提是在核心中有設定 IPFIREWALL_VERBOSE 的選項。有時因爲同樣的封包太多，會使記錄檔保有大量相同的記錄，因此我們會在核心中再設定 IPFIREWALL_VERBOSE_LIMIT 這個選項，來限制要記錄多少相同的封包。

proto：網路協定的名稱，如果使用 ip 或 all 表示所有協定。可以使用的選項有 ip,all,tcp,udp,icmp 等。

src 及 dist：src 是封包來源；dist 是封包目的地。在這二個項目可以用的關鍵字有 any, me, 或是以 <address/mask>[ports] 的方式明確指定位址及埠號。

若使用關鍵字 any 表示使這條規則符合所有 ip 位址。若使用關鍵字 me 則代表所有在本系統介面的 IP 位址。而使用明確指定位址的方式有下列三種：

 IP 位址，指定一個 IP，如 1.2.3.4。

 IP/bits，如 1.2.3.4/24，表示所有從 1.2.3.0 到 1.2.3.255 的 IP 都符合規則。

/27 1/8th of a Class C 32 hosts

/26 1/4th of a Class C 64 hosts

/25 1/2 of a Class C 128 hosts

/24 1 Class C 256 hosts

/23 2 Class C 512 hosts

/22 4 Class C 1,024 hosts

/21 8 Class C 2,048 hosts

/20 16 Class C 4,096 hosts

/19 32 Class C 8,192 hosts

/18 64 Class C 16,384 hosts

/17 128 Class C 32,768 hosts

/16 256 Class C 65,536 hosts (= 1 Class B)

/15 512 Class C 131,072 hosts

/14 1,024 Class C 262,144 hosts

/13 2,048 Class C 524,288 hosts

IP:mask，由 IP 加上子網路遮罩，如 1.2.3.4:255.255.240.0 表示從 1.2.0.0 到 1.2.15.255 都符合。

而在 me，any 及 指定的 ip 之後還可以再加上連接埠編號 (ports)，指定 port 的方法可以是直接寫出 port ，如 23；或給定一個範圍，如 23-80；或是指定數個 ports，如 23,21,80 以逗點隔開。或者是寫出在 /etc/services 中所定義的名稱，如 ftp，在 services 中定義是 21，因此寫 ftp 則代表 port 21。

interface-spec：可以使用下列幾個關鍵字的結合

表20

in	只符合流入的封包。
out	只符合流出的封包。
via ifX	封包一定要經過介面 ifX，if 為介面的代號，X 為編號，如 vr0。
via if*	表示封包一定要經過介面 ifX，if 為介面的代號，而 * 則是任何編號，如 vr* 代表 vr0,vr1,...。
via any	表示封包一定要經過任何介面。
via ipno	表示封包一定要經過 IP 為 ipno 的介面。

via 會使介面永遠都會被檢查，如果用另一個關鍵字 recv ，則表示只檢查接收的封包，而 xmit 則是送出的封包。這二個選項有時也很有用，例如要限制進出的介面不同時：

```
ipfw add 100 deny ip from any to any out recv vr0 xmit ed1
```

recv 介面可以檢查流入或流出的封包，而 xmit 介面只能檢查流出的封包。所以在上面這裡一定要用 out 而不能用 in，只要有使用 xmit 就一定要使用 out。另外，如果 via 和 recv 或 xmit 一起使用是沒有效的。

有的封包可能沒有接收或傳送的介面：例如原本就由本機所送出的封包沒有接收介面，而目的是本機的封包也沒有傳送介面。

options：另外還有一些常用的選項 ，更多選項請 man ipfw：

表21

keep-state	當符合規則時，ipfw 會建立一個動態規則，內定是讓符合規則的來源及目的地使用相同的協定時就讓封包通過。這個規則有一定的生存期限(lift time，由 sysctl 中的變數所控制)，每當有新的封包符合規則時，便用重設生存期限。
bridged	只符合 bridged 的封包。
established	只適用於 TCP 封包，當封包中有 RST 或 ACK bits 時就符合。
setup	只適用於 TCP 封包，當封包中有 SYN bits 時就符合。

以上的說明只是 man ipfw 中的一小部份。如果你想要對 ipfw 更了解，例如如何使用 ipfw 來限制頻寬等，建議你 man ipfw。

不知道您看了這麼多的規則是否覺得眼花撩亂，如果不了解 TCP/IP 的原理，徹底了解 ipfw 的設定還真不容易。沒關係，我們下面將舉幾個簡單、常用的設定，這些範例應該夠平常使用了。

13.3.2 範例

我將原本的 /etc/rc.firewall 備份成 rc.firewal.old，並將它改成下列內容，請注意，這裡只是範例，只供參考：

```
# 設定我的 IP
myip="1.2.3.4"
# 設定對外的網路卡代號
outif="vr0"
# 設定對內的網路上代號
inif="vr1"
#清除所有的規則
/sbin/ipfw -f flush
# Throw away RFC 1918 networks
```

```
${ipfw} add deny ip from 10.0.0.0/8 to any in via ${oif}
${ipfw} add deny ip from 172.16.0.0/12 to any in via ${oif}
${ipfw} add deny ip from 192.168.0.0/16 to any in via ${oif}
# 只允許內部網路對 192.168.0.1 使用 telnet 服務
/sbin/ipfw add 200 allow tcp from 192.168.0.1/24 to 192.168.0.1 telnet
# 拒絕其他人連到 port 23，並記錄嘗試連線的機器
/sbin/ipfw add 300 deny log tcp from any to me 23
# 拒絕任何 ICMP 封包
/sbin/ipfw add 400 deny icmp from any to any

# 下面這台機器是壞人，不讓它進來，並記錄下來
/sbin/ipfw add 1100 deny log all from 211.21.104.102 to any
# NAT 的設定
/sbin/ipfw add divert natd all from any to any via vr0
# 限制內部網域對外下載最大頻寬為 20KBytes/s，上傳最大頻寬為 5KBytes/s
ipfw pipe 20 config bw 20KBytes/s
ipfw add pipe 20 ip from any to 192.168.0.1/24 out
ipfw pipe 30 config bw 5KBytes/s
ipfw add pipe 30 ip from 192.168.0.1/24 to any in
# 允許本機對任何地方連線
/sbin/ipfw add check-state
/sbin/ipfw add 2000 allow udp from ${myip} to any keep-state
/sbin/ipfw add 2100 pass ip from ${myip}  to any

# 允許外界使用郵件服務
/sbin/ipfw add 3000 pass tcp from any to ${myip} 25 in via ${outif}
# 不允許內部的 IP 從外部連進來
/sbin/ipfw add 1200 add deny ip from ${myip}/24 to any in via ${oif}
```

```
# 其他都拒絕，如果沒有在 kernel 中設定
# IPFIREWALL_DEFAULT_TO_ACCEPT 則內定就有下列這一條
/sbin/ipfw 65535 add deny all from any to any
```

存檔後就可以使用 sh rc.firewall 來執行新的規則了。如果您將規則放在 /etc/rc.firewall 中，則開機時會自動執行。

13.3.3 一些小建議

在建立一個封包過濾的防火牆時，應該盡可能阻擋 port 1024 以下的 TCP 服務，例如只通過 SMTP 封包 (port 25) 給郵件伺服器；拒絕所有 UDP 連線 (只有少部份服務如 NFS 會用到)；一些只有內部才會使用的服務，如資料庫等也不必對外開放。

另外，同樣的防火牆限制可以使用不同的語法來展現，應該要試著讓規則數量越少越好，以加快處理速度。

在更新 firewall 規則時，如果規則沒有寫好，而你又是以遠端登入的方式修改規則，很可能會因此無法繼續登入。因此建議更新規則時最好在 console 前執行，若迫不得已一定要使用遠端登入，建議您執行 /usr/share/examples/ipfw/change_rules.sh 這支程式來編輯規則：

cd /usr/share/examples/ipfw

sh change_rules.sh

接著會出現文書編輯軟體並最動載入 /etc/rc.firewall 讓你編輯，結束離開後，會詢問是否要執行更新。如果執行新的規則後造成斷線，它會自動載入舊的規則，讓我們可以再次連線。

13.4 封包過濾橋接器

當我們的內部網路有不同 class 的主機時，例如內部有 140.115.2.3 及 140.115.5.6 這二台電腦時，就無法使用傳統的防火牆。如果要在這二台機器連到網際網路中途中使用防火牆，我們必須使用新的方式，就是這裡要介紹的橋接器。

我們可以使用 FreeBSD 爲橋接器，利用它來做封包過濾的動作，而絲毫不影響內部的主機原本的設定。爲了達到這個功能，我們必需要有二張支援 promiscuous mode 的網路卡，現在的網路卡大部份都有支援。二張網路卡當中，一張需要設定 IP，另一張不需要。至於您要將 IP 設定在哪一張卡都可以，建議是設在對外的網路卡上。

首先，我們必須在核心中加入關於橋接器的設定：

```
# 支援橋接器
options BRIDGE
# 防火牆設定
options IPFIREWALL
options IPFIREWALL_VERBOSE
# 我們這裡不將防火牆預設為接收所有封包
#options IPFIREWALL_DEFAULT_TO_ACCEPT
```

重新編譯核心後，在重開機前，我們先設定一下 /etc/rc.conf：

```
firewall_enable="YES"
firewall_type="open"
```

還有一件事要做，當在乙太網路上跑 IP 協定時，事實上使用二種乙太網路協定，一個是 IP，另一個是 ARP。ARP 協定是當機器要找出給定 IP

位址所對應的乙太網路位址時使用的。ARP 並不是 IP 層的一部份，只是給 IP 應用在乙太網路上運作。標準的防火牆規則中並未加入對於 ARP 的支援，幸運的是，高手們的在 ipfirewall 程式碼中加入了對封包過濾橋接器的支援。如果我們在 IP 位址 0.0.0.0 上建立一個特別的 UDP 規則，UDP 埠的號碼將被使用來搭配被橋接封包的乙太網路協定號碼，如此一來，我們的橋接器就可以被設定成傳遞或拒絕非 IP 的協定。請在 /etc/rc.firewall 中接近文件頂端處理 lo0 的那三行之下(就是有寫 Only in rare cases do you want to change these rules 的地方)加入下面一行：

```
${fwcmd} add allow udp from 0.0.0.0 2054 to 0.0.0.0
```

現在我們就可以重新開機了。重開機之後，先執行下列指令來啓動橋接器：

```
# sysctl -w net.link.ether.bridge_ipfw=1
# sysctl -w net.link.ether.bridge=1
```

現在我們可以將機器放在內外二個網域之間了。因爲我們之前在 /etc/rc.conf 中，設定防火牆完全打開，不阻擋任何封包，所以放在二個網域之間時，運作應該沒有問題。我們之前只設了一張網路上的 IP，而在執行了上述的指令之後，第二張網路卡便開始運作。

下一步就是將我們啓動橋接器的指令放在 /etc/rc.local 中，讓系統在開機時自動執行。或者，我們可以在 /etc/sysctl.conf 中加入下面二行：

```
net.link.ether.bridge_ipfw=1
net.link.ether.bridge=1
```

接下來我們就可以依自己的需求在 /etc/rc.firewall 文件的最後面加上我

們自己想要的防火牆規則了。以下是一個簡單的設定規則，假設橋接器的 IP 是 140.115.75.137，內部有二台主機，一台提供網頁服務，一台是 BBS：

```
us_ip=140.115.75.137
basrv_ip=140.115.3.4
bbs_ip=140.115.5.6
oif=fxp0
iif=fxp1
ipfw="/sbin/ipfw"
# Things that we've kept state on before get to go through in a hurry.
${ipfw} 1000 add check-state

# Throw away RFC 1918 networks
${ipfw} 1100 add deny ip from 10.0.0.0/8 to any in via ${oif}
${ipfw} 1200 add deny log ip from 172.16.0.0/12 to any in via ${oif}
${ipfw} 1300 add deny log ip from 192.68.0.0/16 to any in via ${oif}
# 允許橋接器本身所有想做的連線 (keep state if UDP)
${ipfw} 1400 add pass udp from ${us_ip} to any keep-state
${ipfw} 1500 add pass ip from ${us_ip} to any
# 允許內部網路任何想做的連線 (keep state if UDP)
${ipfw} 1600 add pass udp from any to any in via ${iif} keep-state
${ipfw} 1700 add pass ip from any to any in via ${iif}

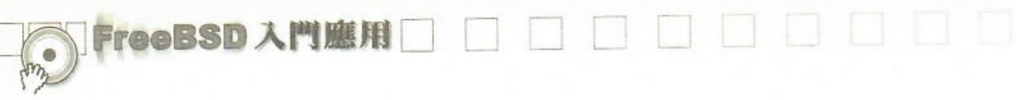
# 允許任何的 ICMP 連線
${ipfw} 1800 add pass icmp from any to any

# 不允許使用 port 888 連線
${ipfw} 2000 add deny log tcp from any to ${bbs_ip} 888
```

```
# TCP section
# 任何地方都可以建立 TCP 連線
${ipfw} 3000 add pass tcp from any to any via ${oif}

# Pass the "quarantine" range.
${ipfw} 3100 add pass tcp from any to any 49152-65535 in via ${oif}
# Pass ident probes. It's better than waiting for them to timeout
${ipfw} 3200 add pass tcp from any to any 113 in via ${oif}
# Pass SSH.
${ipfw} 3300 add pass tcp from any to any 22 in via ${oif}
# Pass DNS. 當內部網路有名稱伺服器時才需要
#${ipfw} add pass tcp from any to any 53 in via ${oif}
# 只傳遞 SMTP 給郵件伺服器
${ipfw} 3400 add pass tcp from any to ${bbs_ip} 25 in via ${oif}
${ipfw} 3500 add pass tcp from any to ${basrv_ip} 25 in via ${oif}
# UDP section
# Pass the "quarantine" range.
${ipfw} 4000 add pass udp from any to any 49152-65535 in via ${oif}
# Pass DNS. 當內部網路有名稱伺服器時才需要
#${ipfw} 4100 add pass udp from any to any 53 in via ${oif}

# 其他的都拒絕
${ipfw} 60000 add deny ip from any to any
```



chapter 14

Proxy Server

14.1 概論

Proxy 可以用來降低 WWW 及 FTP 的網路流量，它的運作原理簡而言之就是當網域中其他電腦要瀏覽網頁時，必須先通過 proxy 伺服器，如果在 proxy 中已存有該網頁時，就直接送出給網域中的電腦；如果沒有，則連到該網頁取回來存放，並送一份給提出要求的電腦，待下次再有相同要求時，就可以直接從 proxy 的存檔中送出。

我們會在 proxy 伺服器中設定資料要存放多久的時間並更新，以免使用者看到的都是存在 proxy 中的舊網頁。這對於內部有大量的使用者時，可以明顯降低網路流量。如果網域中只有十幾台電腦，那麼 proxy 就比較不能發揮它的功效了。我們也可以在 proxy 中設定使用者不可以瀏覽的網頁，限制使用者連到某些網站。

一般而言，使用者要使用 proxy 時，必須先在自己電腦上做一些設定。我們也可以利用 proxy 結合防火牆，當使用者對外提出 HTTP 要求時，即自動轉向到 proxy server，如此一來，使用者端便不需做任何設定，甚至不會發覺有 proxy 的存在。

在這裡我們介紹一個被廣為使用的 proxy 軟體 「squid」。squid 的安裝及設定很簡單，我們會在下面幾節中一一為大家介紹。

14.2 安裝 Squid

感謝美好的 FreeBSD ports，可以讓我們很簡單的安裝 squid。請使用下列指令：

```
# cd /usr/ports/www/squid24
# make install
```

如此便已安裝 squid 了。接下來就必須要修改 /usr/local/etc/squid/squid.conf 了，squid.conf 是整個 squid 的設定所在，內容很多，我們下一章再做詳細的介紹。這裡我們只作簡單的設定，也就是先定義誰可以使用 proxy，例如我要定義只有 192.168.0.* 及 *.alexwang.com 可以使用，則在 squid.conf 中找到 ACCESS CONTROLS 的區段中的 acl 部份，加入下面內容：

```
acl domain_allowed srcdomain .alexwang.com
acl ip_allowed src 192.168.0.0/24
```

在上面二行中，我們先將網址及 IP 定義一個名字，將所有在 *.alexwang.com 的網址定義名稱爲 domain_allowed；將 192.168.0.0-192.168.0.255 的 IP 命名爲 ip_allowed。接著再找到 http_access 的部份，在 http_access deny all 之前加入下面有內容：

```
http_access allow domain_allowed
http_access allow ip_allowed
```

上面是允許名稱爲 domain_allowed 及 ip_allowed 的網址使用 proxy。關於上述設定的說明請參考 squid.conf 介紹。

修改完 squid.conf 之後，接著要建立 cache 目錄結構，預設的 cache 目

錄在 /usr/local/squid/chache 中，如果要將它放在其它目錄的話，必須修改 squid.conf 並建立所設定的目錄。接著使用下列指定來改變檔案權限並建立 cache 的目錄結構：

```
# chown -R nobody.nogroup /usr/local/squid/cache
# chown -R nobody.nogroup /usr/local/squid/logs
# /usr/local/sbin/squid -z
```

當下達 squid -z 的指令時，必須等上幾分鐘的時間。安裝好 squid 之後，在 /usr/local/etc/rc.d/ 有一個名爲 squid.sh 的檔案，表示在開機時便用自動啓動 squid。當做好一切設定之後，我們只要執行下列指令即可啓動 squid。

```
# /usr/local/etc/rc.d/squid.sh start
```

我們在這裡的設定只是讓你能立即使用 proxy，詳細的設定必須再修改 squid.conf，一些相關的設定如某個時段禁止連到某個位址、資料保存期限等等。

接下來是 client 端的設定，以 MS Windows 爲例：

對著桌面的 IE 按右鍵選 [內容] 或是在控制台中選 [網際網路選項]，出現圖14-1的視窗：

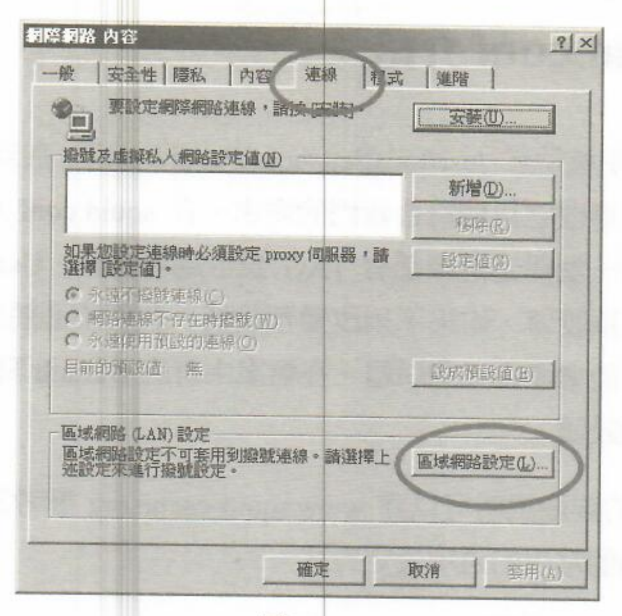


圖 14-1

選擇上方 [連線] 的標籤後，再點選 [區域網路設定]，出現下面視窗。接著在Porxy 伺服器的部份，在網址的部份輸入 proxy server 的網址，連接埠預設值是 3128，接著按確定即可使用瀏覽器來試試看可不可以使用了：

區域網路 (LAN) 設定
自動組態
自動設定會取代手動設定。要確保使用手動設定，請停用自動設定。
自動偵測設定(A)
使用自動組態指令碼(S)
網址(R)
Proxy 伺服器
在您的區域網路使用 Proxy 伺服器 (這些設定將不會套用到撥號或 VPN 連線)(X)
網址(E): 192.168.0.1 連接埠(T): 3128 進階(C)...
近端網址不使用 Proxy(B)
確定 取消

圖 14-2

14.3 Squid.conf 介紹

Squid 的設定檔位於 /usr/local/etc/squid/squid.conf。在安裝完 Squid 之後，我們必須修改它以期符合我們的需求。在 squid.conf 中，開頭爲 "#" 表示註解，每一個設定選項都有 TAG 表示選項名稱、Usage 表示用法、Defaualt 表示預設值。如果不須改變預設值，我們不必將該行的註解 "#" 拿掉，否則可能會產生一些問題。在檔案中有的 Default 寫著 "none" 表示該選項沒有預設值。

squid.conf 的詳細說明可以到 www.squid-cache.org 參考設定手冊。我們在這裡介紹幾個常用的選項設定方式：

選項名稱： http_port

用法： port

hostname:port

1.2.3.4:port

這是 Squid 要接收 HTTP 要求時所使用的 port。可以使用三種格式：只指定 port、使用 hostname 及 port、或是 IP 位址及 port。一般而言，我們只要指定 port 即可。我們也可以設定多行 http_port 來使用多個 port。

預設的 port 是 3128

我們也可以在命令列下執行 squid 這個指令，並以參數 -a 來取代 squid.conf 中所設定的第一個 port。例如，我們要啓動 squid 並將 port 改成 8080，可以下指令： /usr/local/sbin/squid -a 8080 使用 -a 來覆蓋在 squid.conf 中所設定的第一個 port 只能適用於只指定 port 時，如果在

port 前有加上 ip 或 hostname 的話，這個指令就不會產生做用。

選項名稱： icp_port

當 Squid 接收到 ICP 查詢時，要回應時所使用的 port，可以設為 0 來停用 ICP 查詢。也可以在命令列使用 -u 來覆蓋這裡的設定。 ICP 是 Squid cache 之間所使用的協定，用來交流多台 Squid 的 cache。目前 ICP 是使用 UDP 協定。

預設值： icp_port 3130

選項名稱： mcast_groups

這個選項是用來指定這台伺服器要加入 ICP 廣播的群組，也就是指定要收到的 ICP 查詢的來源主機。 請注意！這個設定是 "接收" 查詢而不是 "送出" 查詢。如果要送出 ICP 廣播查詢是使用 cache_peer。 不可以將已加入別的群組的主機加入。

如果你想更深入了解 multcast 請參考 Squid FAQ (http://www.squid-cache.org/FAQ/)。

用法： mcast_groups 239.128.16.128 224.0.1.20

預設是不加入任何群組。

預設值：無

選項名稱：cache_peer

用法：cache_peer hostname type http_port icp_port [options]

這個選項可以讓你設定上游的 proxy server，當本地沒有該筆資料時，則向上游查詢。例如上游的 proxy 是 proxy.ncu.edu.tw，我們可以設定：

cache_peer proxy.ncu.edu.tw parent 3128 3130

選項名稱：no_cache

設定不需要使用 cache 的項目。例如 CGI 不使用 cache，則可以做下列的設定：

acl QUERY urlpath_regex cgi-bin \?

no_cache deny QUERY

如果連 php、asp 都不要做 cache：

acl QUERY urlpath_regex cgi-bin \? \.php \.asp \.cgi

no_cache deny QUERY

選項名稱：cache_mem

設定存放在記憶體中的資料大小，以 byte 爲單位。這並不表示 squid 程式將在你記憶體中的所佔的容量，除了 cache 的資料外，還有一些額外的東西會被放到記憶體中 (如 HD 的 index)。因此實際使用的記憶體大小可能是這裡設定的二倍或三倍。建議設定爲實際記憶體的三分之一。例如：

cache_mem 64 MB

選項名稱：maximum_object_size

cache 物件的大小，超過此設定值則不存放於硬碟中。例如要設定檔案大於 8 MB 則不儲存：

maximum_object_size 8192 KB

選項名稱：cache_dir5

設定 cache 在硬碟中的目錄、大小限制等。預設值是

cache_dir ufs /usr/local/squid/cache 100 16 256

ufs 是 squid 的儲存格式，一定要加。接下來的 /usr/local/squid/cache 是存放 cache 資料的地方。100 表示最多使用 100 MB 做爲 cache 的空間，我們可以把它調大一點。而 16 256 是 cache 目錄第一層及第二層的結構，建議不要更動。之前使用 /usr/local/sbin/squid -z 就是在建立 cache 目錄的結構。

爲了讓 cache 的資料能存放多一點，我們可以將設定改成下面這樣：

cache_dir ufs /usr/local/squid/cache 2000 16 256

選項名稱：cache_access_log

cache 的使用記錄存放的位置，內容包含了所有 HTTP 及 ICP 要求。預設值是：

cache_access_log /usr/local/squid/logs/access.log

選項名稱：cache_log

這是設定另一個記錄檔的位置，該檔中包含了 cache server 的一些資訊、和 cache_peer 之間的連線等等。我們可以更改 "debug_options" 的選項來設定要記錄的資訊。預設值是：

cache_log /usr/local/squid/logs/cache.log

選項名稱：cache_store_log

記錄哪些資料被儲存。可以使用 "none" 來停止這方面的記錄。上面的 log 檔，都可以用一些工具來分析 squid 的使用情形，但 store log 並沒有，所以如果不想記錄可以使用 "none" 來停止。

選項名稱：pid_filename

設定存放 pid (process id) 的檔案位置。

選項名稱：ftp_user

設定當 proxy 以 anonymous FTP 連線時，要使用的 email。你可以設爲：

ftp_user you@yourdomain.com

選項名稱：cache_dns_program

squid 使用的 dns 查詢程式的位置。squid 在查詢 dns 時，會新增一個子程序 (process child) 來做查詢，以免查詢時間太長阻礙了 squid 的正常運作。

選項名稱：dns_children

設定 dns 查詢的 children 最大數量，預設值是 5，最大可以設爲 32。以一個忙錄的 squid server 而言，最小建議設爲 10。

選項名稱：acl

Access Control List。我們可以利用它來控制連線的權限及狀態。用 acl 來搭配 http_access 等，可以讓我們更方便管理。

用法：acl aclname acltype string1 ...
acl aclname acltype "file" ...

aclname 是我們自己命名的識別字，要注意不要使用到 squid 的關鍵字。而 acltype 可以使用下列的選項：

src：表示來源，即 client 的 IP。

如：acl localip src 192.168.0.0/24

srcdomain：表示來源，爲 domain name 的格式。

如：acl localdomains srcdomain .alexwang.com

dst：表示目的地的 IP。

如：acl acceleratedhost dst 172.16.1.115/255.255.255.255

dstdomain：表示目的地的 domain name。

如：acl badwebsite dstdomain www.sex.com

srcdom_regex：對來源的 URL 做正規運算式(regular expression)運算。

如：acl localwebsite srcdom_regex mydomain.com

dstdom_regex：對目的地的URL 做正規運算式(regular expression)運算。

如：acl sexwebsite srcdom_regex ^http://www.playboy.com

time：指定時間。

用法：acl aclname time [day-abbrevs] [h1:m1-h2:m2]

day-abbrevs:

S - Sunday

M - Monday

T - Tuesday

W - Wednesday

H - Thursday

F - Friday

A - Saturday

h1:m1 一定要比 h2:m2 小

port：指定連接埠。

如：acl SSL_ports port 443

proto：指定所使用的通訊協定。

如：acl allowprotolist proto HTTP

更多的 acltype 關鍵字請看 squid.conf。

選項名稱：http_access

用來控制要開放 squid 給哪些來源使用。預設是全部拒絕，最好不要不設防，例如使用 http_access allow all 就是一個不好的方式。

我們可以利用 http_access 搭配 acl 來控制使用情形。例如要開放給內部網域使用：

```
# INSERT YOUR OWN RULE HERE TO ALLOW ACCESS FROM
# YOUR  CLIENTS

acl domain_allowed srcdomain .alexwang.com
acl ip_allowed src 192.168.0.0/24
http_access allow domain_allowed
http_access allow ip_allowed
```

在設定了可以使用的來源之後，最後再加上：

http_access deny all

來拒絕其他連線。

選項名稱：icp_access

控制 squid 只回應哪些 sibling/child 的 ICP 詢問。例如，只允許校內機器使用：

acl ncu src 140.115.0.0/255.255.0.0

icp_access allow ncu

icp_access deny all

選項名稱：miss_access

控制只能做為 sibling 而不做 parent，例如：

acl localclients src 172.16.0.0/16

miss_access allow localclients

miss_access deny !localclients

選項名稱：cache_peer_access

和 cache_peer 一樣，只是這個選項可以使用 acl 來控制。

選項名稱：cache_mgr

squid 管理者的 email。可以設定為本機的使用者：

cache_mgr webmaster

或別台主機的 email：

cache_mgr jack@otherdomain.com

選項名稱：cache_effective_user

選項名稱：cache_effective_group

如果 squid 是以 root 的身份來執行，它會自動切換成這裡所設定的使用

者及群組。預設是 nobody，所以我們才會把 /usr/local/squid/cache 及 /usr/local/squid/logs 的擁有者改成 nobody。

選項名稱：visible_hostname

設定當 error message 顯示時的 hostname。如果沒有設定則以 gethost-name() 所得到的 hostname 爲主。

選項名稱：httpd_accel_host

選項名稱：httpd_accel_port

如果要以 Transparent Proxy 來執行，則設定：

httpd_accel_host virtual

httpd_accel_port 80

選項名稱：httpd_accel_with_proxy on|off

如果要以 Transparent Proxy 及一般的 proxy 來執行則設爲 on。

選項名稱：httpd_accel_uses_host_header on|off

如果要以 Transparent Proxy 來執行，則設爲 on。讓 squid 可以經由 HTTP header 來判斷 url。

選項名稱：logfile_rotate

設定要保留的 log file 份數，讓執行 squid -k rotate 來分析 squid 時使用。

選項名稱：err_html_text

該你可以在指定要出現的 error page (就是會有 mailto 的網頁)。

選項名稱：deny_info

用法：deny_info err_page_name acl

可以讓你指定當拒絕連線時要使用的 HTML 檔案。例如：

deny_info ERR_CUSTOM_ACCESS_DENIED bad_guys

這些 HTMl 檔都放在 /usr/local/etc/squid/errors/，而且沒有副檔名 .html。

選項名稱：error_directory

如果自行建立一些 error message 的 HTML，我們可以放在預設的 /usr/local/etc/squid/errors 或是自行指定目錄位置。

14.4 Transparent Proxy

有的使用者可能不想設定 Proxy，或者是管理者希望能不必設定 client 端即可使用 proxy，我們可以使用 Transparent Proxy 來強迫使用者使用 Proxy。只要在 Gateway 上使用防火牆來將對外的連線要求重導到 proxy server 即可。

這裡我們假設 Proxy 就是 Gateway，它身兼 NAT 功能，我們內部的網域為 192.168.0.1/24。我們不提 NAT 的設定，只針對 Transparent Proxy 相關的設定加以說明。首先要先確認 kernel 中除了原有關於防火牆的設定外，有沒有 FORWARD 封包的設定，如果沒有必需自行加入並重編 kernel：

```
# 原有的防火牆設定
options IPFIREWALL
options IPDIVERT
options IPFIREWALL_VERBOSE
options IPFIREWALL_DEFAULT_TO_ACCEPT

# 新加入關於封包轉向的設定
options IPFIREWALL_FORWARD
```

接著在 /etc/rc.firewall 中加入：

```
/sbin/ipfw add 50000 fwd 127.0.0.1,3128 tcp from 192.168.0.0/24 to any 80
```

上面那一行的設定表示凡是由 192.168.0.* 的 IP 要連到任何對於的 port 80 時，便轉向到 127.0.0.1 這台機器(也就是本機) 的 port 3128。你可以依自己的情況加以修改。然後編輯 /usr/local/etc/squid/squid.conf，在最開頭加入下面這幾行：

```
httpd_accel_host virtual
httpd_accel_port 80
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

完成後重新啓動 squid 及 ipfw 即可使用。

14.5 Proxy 管理

14.5.1 log 檔移轉

log 檔在頻繁的使用下會一直成長，因此我們可以利用 crontab 來設定每天備份各個 log 檔。執行 crontab -e 後，加入下列內容：

```
0 5 * * * /home/squid/bin/squid -k rotate
```

這裡我們設定為每天早上五點備份 log 檔，如果您對於 crontab 的用法不熟悉，請 man crontab。

14.5.2 關機注意事項

由於 squid 對於硬碟的讀寫十分頻繁，而且有大量的資料在記憶體中。因此在關機前要先停止 squid：

```
# /usr/local/etc/rc.d/squid.sh stop
```

chapter 15 資料庫系統

15.1 概論

電子商務的興起讓資料庫的應用更受到大家的矚目。在資訊科學的應用上，資料庫可以說是最歷久彌堅的領域。近來，資料產生和資料收集方面的技術有非常快速的進展。許多商業產品廣泛使用了條碼、許多企業和政府的交易皆已電腦化，這使得電腦成爲資料收集的主要工具。同時，數以百萬計的資料庫正被使用在企業管理、政府管理、科學和工程的資料管理和許多其它的應用上。

我們可以安裝一套資料庫系統，並經由一個介面自行開發程式來使用它。資料庫的好處有很多，相信對資料庫稍有涉入的人都知道，例如資料存取快速、不重覆、權限控制、資料獨立性等等。以寫一個簡單的留言版程式而言，傳統上使用檔案做爲留言的記錄，若要刪除一筆資料，必須對整個檔案一行一行的比對；但資料庫只需指定該留言的編號即可。不過，如果把資料庫系統局限於留言版也太大才小用了。

我們將介紹在 FreeBSD 上使用資料庫，因爲目前網頁資料庫使用情形十分風行，尤其在網頁開發上使用 MySQL+PHP 更是絕配，所以我以 MySQL 爲主介紹它的用法。爲什麼選用 MySQL 而不選擇其他的資料庫，因爲它簡單、免費、功能強大、具有多平台、多執行緒、且參考文件多。你可以到 MySQL 的網站上參觀 http://www.mysql.com。除了 MySQL 外，還有另一套不錯的資料庫 Postgre SQL 也不錯。在這一章中，我們不再介紹如何安裝 MySQL，您可以參考第十章網頁伺服器中關於安裝 MySQL 的說明。

在使用資料庫之前，我們必須先了解一些簡單而基本的資料庫理論。基本上資料庫的結構有下列幾個特點：

- 一個資料庫系統中可以有多個獨立資料庫
- 資料庫是由許多資料表 (table) 所組成
- 資料表中包含許多記錄 (record)。
- 每一筆記錄中的欄位數目都一樣。
- 每一個欄位儲存一種分類過的資料。

例如我們有一個資料庫名稱是NCU，其中有多個資料表，其中一個資料表名為 student 內容如下：

表22

STUDENT_ID	LAST_NAME	FIRST_NAME	DEPARTMENT
1	Chang	Jack	MIS
2	Wang	Alex	BA

在資料表中有許多欄位 (column)，每個欄位都有一個名稱，也就是第一列 (row) 中的 STUDENT_ID、LAST_NAME、FIRST_NAME、DEPARTMENT。接著我們將資料存入，每一筆記錄我們都可以看成一列 (row)，每一個記錄都有一個「唯一的 ID (編號)」。唯一的 ID 十分重要，它是我們在存取資料庫時的依據。在新增資料時，以 MySQL 而言，我們可以自行指定 ID 或是由系統自行取得。

另一個觀念是關聯式資料庫。關聯式資料庫的意義就是每一個資料表間可以存在關係，例如我們在 NCU 的資料庫中有另一個資料表名為 score，內容如下：

表23

SCORE_ID	STUDENT_ID	CHINESE	ENGLISH
1	2	99	90
2	1	89	87

score 資料表中存放學生的成績，我們不需在該資料表中存放學生的資訊，只要在該資料表中存放一個欄位名為 STUDENT_ID，經由這一個唯一的 ID 我們可以去 studnet 的資料表中找到學生的資料。有了這些觀念就足以讓我們開發出許多資料庫的程式了。

15.2 SQL 語法介紹

SQL (Structured Query Language) 語法十分簡單，它是關聯式資料庫的標準語言，雖然在某些不同資料庫系統上有些許的差異，但基本上都遵循一定的標準。

我們可以在命令列下進入 MySQL 來練習 SQL 的語法：

```
# /usr/local/mysql/bin/mysql -u root -p
```

```
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 14 to server version: 3.23.46

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

下完指令後會要求輸入密碼， 之後你就可以直接在出現的命令列 mysql> 之後輸入 SQL 的語法了。



關於 MySQL 詳細的語法，你可以參考 MySQL 中文參考手冊，該文件可以在 http://cnpa.yzu.edu.tw/~cfc/docs/mysqldoc_big5/manual_toc.html 中找到。該文件中對於 MySQL 每個細節都有詳細的描述，例如欄位的名稱限制、規則等，我們不會在這裡提及。我們只介紹幾個簡單而常用的指令。我們以建立一個學生資料表來說明這些語法。

- CREATE
- ALTER
- DROP
- INSERT
- SELECT
- UPDATE
- DELETE
- 製成 script 檔

15.2.1 CREATE

建立資料庫：CREATE DATABASE db_name
建立資料表：CREATE TABLE tbl_name [(create_definition,...)] [options]

我們先建立一個名爲 NCU 的資料庫：

```
mysql> CREATE DATABASE NCU;
```

請注意，每一個指令皆以 ";" 爲結尾，如果沒有 ";" 就算換行也是代表同一條指令的延續。

我們可以使用下列指令 show 來列出系統中已存在的資料庫有哪些：

```
mysql> show databases;
+-----------+
| Database |
+-----------+
| mysql      |
| test          |
| NCU         |
+-----------+
3 rows in set (0.01 sec)
```

接著用 USE 這個指令來使用 NCU 資料庫：

```
mysql> USE NCU;
```

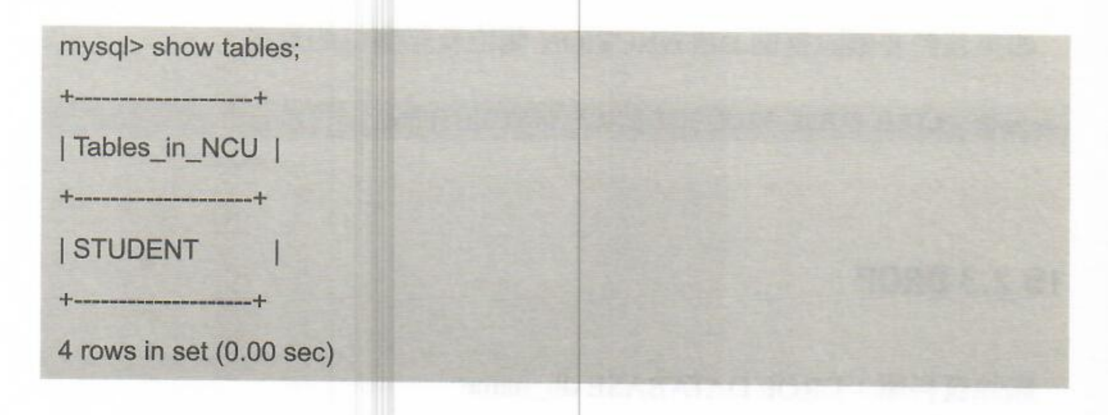
接著建立一個放置學生資料的資料表，名為 STUDENT，內容為編號 (STUDENT_ID)、姓名 (NAME)、科系 (DEPARTMENT)：

```
mysql> CREATE TABLE STUDENT (
STUDENT_ID int(10) DEFAULT '0' NOT NULL AUTO_INCREMENT,
NAME varchar(50),
DEPARTMENT varchar(10),
PRIMARY KEY (STUDENT_ID) );
```

在上面的指令中，我們定義學生編號為十位數的整數(int)，內定值是 0，不可以是空的 (NOT NULL)，數字自動增加 (AUTO_INCREMENT)。姓名是最長為五十個字節的字串(VARCHAR)，科系為最長十個字節的字串。最後定義主要的 id 是 STUDENT_ID，也就是該資料表中的唯一 ID。

我們可以看到在建立資料表時，我們會順便劃分各個欄位所要儲存的資料長度及其格式，常用的欄位格式請參考 MySQL 中文參考手冊。

如果要看現在使用的資料庫中有哪些資料表，一樣可以使用指令 show 來查看：



```
mysql> show tables;
+--------------------+
| Tables_in_NCU  |
+--------------------+
| STUDENT        |
+--------------------+
4 rows in set (0.00 sec)
```

15.2.2 ALTER

建立了資料表後，如果發現資料表的欄位不符需求，我們不必將資料表刪除重建，可以使用 ALTER 指令來修改資料表的格式。另如我們要新增一個姓別欄位，內容只可以是 "男" 或 "女"，我們可以使用下面的指令：

```
mysql> ALTER TABLE STUDENT
ADD SEX ENUM('男','女') DEFAULT '女';
```

我們增加了一個欄位 SEX，使用 ENUM 的格式，指定內容只能為 "男" 或 "女"，且預設值是 "女"。

如果我們要將 SEX 欄位改名為 DISTINCTION，並將格式改為 VARCHAR：

```
mysql> ALTER TABLE STUDENT
CHANGE SEX DISTINCTION VARCHAR(4);
```

如果我們只是要將 SEX 欄位格式改為 VARCHAR，但不更改名稱，只要將上面的指令中 DISTINCTION 改成 SEX 即可。

如果我們要刪除整個 DISTINCTION 欄位及該欄位的資料：

```
mysql> ALTER TABLE STUDENT DROP DISTINCTION;
```

15.2.3 DROP

刪除資料庫：DROP DATABASE db_name
刪除資料表：DROP TABLE table_bame

我們可以使用 DROP 指令來刪除不要的資料。例如我們要刪除 STUDENT 這一個資料表的話，可以使用下列指令：

```
mysql> DROP TABLE STUDENT;
```

15.2.4 INSERT

使用 INSERT 指令可以讓我們一筆一筆增加資料。

表24

STUDENT_ID	NAME	DEPARTMENT
1	JACK	MIS

假設我們的資料表中的欄位如上表，我們要新增一筆資料，姓名是 JACK、部門是 MIS：

```
mysql> INSERT INTO STUDENT (NAME, DEPARTMENT)
VALUES ('JACK', 'MIS');
```

由於我們在指定 STUDENT_ID 的格式時，加了參數 AUTO_INCREMENT，所以我們不需指定值，mysql 會自動幫我們依序指定。

15.2.5 SELECT

我們可以使用 SELECT 來看資料表中的資料，還可以依自己給定的條件來過濾資料。

假設我們要看 STUDENT 資料表中的所有資料的話，可以使用下列指令：

```
mysql> SELECT * FROM STUDENT;
```

假設我們只要看 NAME 及 DEPARTMENT 欄位的話，我們可以使用下列指令：

```
mysql> SELECT NAME, DEPARTMENT FROM STUDENT;
```

假設我們只要看 NAME 欄位，而且所屬部門為 MIS 的人：

```
mysql> SELECT NAME FROM STUDENT
WHERE DEPARTMENT='MIS';
```

假設我們要看 MIS 部門中的人所有欄位，而且輸出結果時要依 STUDENT_ID 來排序：

```
mysql> SELECT * FROM STUDENT
WHERE DEPARTMENT='MIS'
ORDER BY STUDENT_ID DESC;
```

最後的 DESC 表示遞減 (descending)，由大到小排序。也可以使用 ASC 來表示遞增 (ascending)。

除了這些之外，在 MySQL 中還有一些可以使用的函式，例如我們可以使用 count() 這個函式來計算出有多少筆記錄：

```
mysql> SELECT count(*) FROM STUDENT WHERE DEPARTMENT='MIS';
+----------+
| count(*) |
+----------+
| 5 |
+----------+
1 row in set (0.00 sec)
```

上述結果表示部門為 MIS 的記錄有五筆。

會了這些基本的 INSERT 指定就夠我們做一般的應用了。

15.2.6 UPDATE

我們可以使用 UPDATE 指令來更新記錄。例如我們要將所有記錄的部門資料為 MIS 者都改成 CSIE，可以使用下列指令：

```
mysql> UPDATE STUDENT SET DEPARTMENT='CSIE'
WHERE DEPARTMENT='MIS';
```

15.2.7 DELETE

DELETE 指令可以讓我們刪除一筆或多筆資料。例如我們要刪除 STUDENT 資料表中姓名為 JACK 的記錄：

```
mysql>  DELETE FROM STUDENT WHRE NAME='JACK';
```

如果我們要刪除姓名為 JACK 且部門為 MIS 的資料：

```
mysql> DELETE FROM STUDENT
WHERE NAME='JACK' AND DEPARTMENT='MIS';
```

15.2.8 製成 script 檔

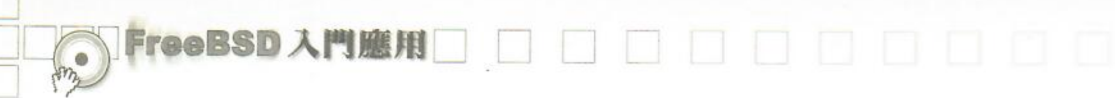
我們可以將要執行的指定製成檔案，以利管理。例如我們寫了一個程式，需要先在資料庫中建立一些資料，我們可以將對資料庫的規劃做成一個檔案來管理。這樣可以使用們要安裝程式時更快速方便。

假設我們要建立一個資料庫 NCU，該資料庫中有一個資料表 STUDENT，資料表中要先建有以下記錄：

表25

STUDENT_ID	NAME	DEPARTMENT
1	JACK	MIS
2	MARY	CSIE

我們先建立一個檔案名為 ncu.sql，內容如下：




```
CREATE DATABASE NCU;
USE NCU;

CREATE TABLE STUDENT (
STUDENT_ID int(10) DEFAULT '0' NOT NULL AUTO_INCREMENT,
NAME varchar(50),
DEPARTMENT varchar(10),
PRIMARY KEY (STUDENT_ID) );

INSERT INTO STUDENT (NAME, DEPARTMENT)
VALUES ('JACK', 'MIS');

INSERT INTO STUDENT (NAME, DEPARTMENT)
VALUES ('MARY', 'CSIE');
```

接著使用下列指令來快速建立資料庫：

```
# /usr/local/mysql/bin/mysql -u root -p <ncu.sql
```

輸入使用者 root 的密碼後就完成建立了。

如果我們在資料庫中早就有一個資料庫名為 NCU，而我們要新增上述資料表及記錄，我們只要將原本 ncu.sql 的內容最前面二行刪除，改成下列內容：

```
CREATE TABLE STUDENT (
STUDENT_ID int(10) DEFAULT '0' NOT NULL AUTO_INCREMENT,
NAME varchar(50),
DEPARTMENT varchar(10),
```

```
PRIMARY KEY (STUDENT_ID) );

INSERT INTO STUDENT (NAME, DEPARTMENT)
VALUES ('JACK', 'MIS');

INSERT INTO STUDENT (NAME, DEPARTMENT)
VALUES ('MARY', 'CSIE');
```

之後再以下列指令來在 NCU 資料庫中建立資料表：

```
# /usr/local/mysql/bin/mysql -u root -p NCU <ncu.sql
```

在網路上有許多 PHP 程式可以下載，下載後要安裝資料庫時，大多是以這種方式來使用。

15.3 MySQL 管理

15.3.1 維護密碼安全

當我們要使用 MySQL 時，必須輸入密碼。輸入密碼的方式有很多種，第一種也是最不安全的一個方式是直接在命令列打指令時就輸入：

```
# /usr/local/mysql/bin/mysql -u root -pmypwd
```

上面這種方法會讓別的使用者使用 ps 指令就可以看到你在執行的指定及密碼。因此絕對不要使用這種方法，請改用下列方式輸入：

```
# /usr/local/myqsl/bin/mysql -u root -p
```

接著會要求你輸入密碼時再輸入即可。

另一個方式是在你的家目錄下建立一個存放密碼的檔案，檔名為 .my.cnf，當 mysql 需要使用密碼時會自動去讀取。該檔的內容如下：

```
[client]
password=your_passowrd
```

接著要把 .my.cnf 的權限改成只有檔案擁有者才可以讀寫：

```
# chmod 600 ~/.my.cnf
```

15.3.2 備份資料庫

資料庫的資料要定時備份，這樣才不會在失手時或系統有問題時產生困擾。在 MySQL 中提供一個備份程式 msqyldump。

假設我們有一個資料庫名為 WWW，我們可以使用下列指令來備份整個資料庫：

```
# /usr/local/mysql/bin/mysqldump -u root -p WWW>www.sql
```

這樣就可以把資料庫的資料存在 www.sql 這個檔案中了。日後要回復時只要使用下列指定就可以把資料存回：

```
# /usr/local/mysql/bin/mysql -u root -p WWW<www.sql
```

我們要注意的是備份出來的檔案應該要放在不同的電腦中，而且要注意權限的控制。由於該檔是文字檔，任何人都可以讀，所以要特別注意。

我們可以利用 crontab 這個指令來定時備份資料庫。我們先建立一個 shell script 檔，名為 backupsql.sh，內容如下：

```
/usr/local/mysql/bin/mysqldump -uroot WWW>/home/www.sql
chmod 600 /home/www.sql
```

接著將該檔權限改成只有擁有人可以讀、寫、執行，其他人都不行：

```
# chmod 700 backupsql.sh
```

然後建立 ~/.my.cnf 檔案內容如下：

```
[client]
password=your_passowrd
```

接著要把 .my.cnf 的權限改成只有檔案擁有者才可以讀寫：

```
# chmod 600 ~/.my.cnf
```

接著要讓它能定時執行，命令列打 crontab -e 來進入文字編輯，加入下列內容：

```
#每天 3:05 備份網頁資料庫
5 3 * * * /root/backupsql.sh
```

15.3.3 使用者管理

MySQL 對於使用者的權限控制存放在名為 mysql 資料庫中的 user 資料表內。user 資料表內有下列幾個欄位：

表26

欄位名稱	權限	說明
Host		使用者來源主機
User		使用者名稱
Password		密碼
Select_priv	select	對 table 做 select 動作
Insert_priv	insert	對 table 做 insert 動作
Update_priv	update	對 table 做 update 動作
Delete_priv	delete	對 table 做 delete 動作
Index_priv	index	對 table 做 index 動作
Alter_priv	alter	對 table 做 alter 動作
Create_priv	create	對 table 、indexs 或 database做 create 動作
Drop_priv	drop	對 table 或 database 做 drop動作
Grant_priv	grant	對 table 或 database 做 grant動作
References_priv	references	對 table 或 database 做 references 動作
Reload_priv	reload	系統管理，權限擁有者可以執行reload, refresh, flush-privileges, flush-hosts,flush-logs, flush-tables
Shutdown_priv	shutdown	系統管理，權限擁有者可以執行shutdown
Process_priv	process	系統管理，權限擁有者可以執行processlist, kill
File_priv	file	對系統上的檔案有存取權限

我們在新增一個 MySQL 使用者時，有二種方式。比較差的方式是使用 INSERT 指令：

```
# /usr/local/mysql/bin/mysql -u root -p mysql
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 224 to server version: 3.23.49

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> insert into user values ('host','user',
password('密碼'),'y','y','y','y','y','y','y','y','y','y','y','y','y','y');
Query OK, 0 row affected (0.00 sec)
```

上面 mysql 指令中的 host 就是要予許連線的主機，如果是本機則輸入 localhost；而 user 是使用者名稱；密碼是該使用者的密碼，要使用 password() 函數來將它加密；接下來的一堆 'Y' 就是對上表中的權限是否要開放，如果不開放則填 'N'。

第二種方式是使用 GRANT 指令來新增使用者，GRANT 在設定使用者權限時，如果使用者存在則更新其權限，如果不存在則新增該使用者：

用法：GRANT 權限 ON 資料庫(或表) TO user@host IDENTIFIED BY '密碼'

範例一：新增一個本機的使用者 admin，並開放所有權限，密碼為 mypwd：

```
mysql> GRANT ALL PRIVILEGES ON *.* TO admin@localhost IDENTIFIED BY
'mypwd';
Query OK, 0 row affected (0.00 sec)
```

範例二：新增一個來自 www.mydomain.com 的使用者 www，並設定只能對 www 資料庫中所有資料表執行 SELECT, INSERT, UPDATE, DROP, CREATE, DELETE, INDEX，密碼為 mypwd：

```
mysql> GRANT SELECT, INSERT, UPDATE, DROP, CREATE, DELETE, INDEX ON
www.* TO www@www.mydomain.com IDENTIFIED BY 'mypwd';
Query OK, 0 row affected (0.00 sec)
```

如果要刪除使用者上述新增的使用者 www，可以使用下列指令：

```
mysql>DELETE FROM user WHERE user='www' and host='www.mydomain.com';
Query OK, 1 rows affected (0.01 sec)
```

在新增或刪除使用者後，離開 MySQL 之前都必須指行下列指令來讓它生效：

```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.01 sec)
```

15.3.4 如何更改使用者密碼

我們可以使用下列指令來更改自己的密碼：

```
# /usr/local/mysql/bin/mysqladmin -u root -p password newpwd
```

上面指令中的使用者是 localhost 的 root ，新的密碼是 newpwd。在輸入指令後，會先詢問你舊的密碼。

我們也可以使用具有管理使用者權限的 mysql 使用者登入 MySQL 後，使用 UPDATE 指令來更改密碼：

```
# /usr/local/mysql/bin/mysql -u root -p mysql
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 224 to server version: 3.23.49
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> UPDATE user set password=password('新密碼')
where user='使用者' and host='主機';
```

chapter 16

Samba 網路芳鄰

16.1 安裝 Samba

在 MS-windows 系統中，我們可以使用 "網路上的芳鄰"。而 FreeBSD 中也有軟體可以讓你在 windows 的網路芳鄰中看到 FreeBSD，甚至可以讓 FreeBSD 存取 windows 的網路芳鄰資料。這就是 Samba 這套軟體的功能。

Samba 的安裝設定很簡單，我們可以使用 ports 來安裝：

```
# cd /usr/ports/net/samba
# make install clean
```

安裝完後，組態檔的範本會放在 /usr/local/etc/smb.conf.default，我們可以直接複製它來加以修改：

```
# cd /usr/local/etc/
# cp smb.conf.default smb.conf
```

Samba 的組態設定除了可以使用文字編輯軟體來修改 smb.conf 外，我們還可以使用瀏覽器連到 Samba 以圖形化介面來設定。

我們先來介紹一下 smb.conf 檔案的內容，在 smb.conf 檔案中行首為 ";" 或 "#" 都是註解。我們可以 man smb.conf 來讀取設定說明。修改完 smb.conf 之後，我們可以使用指令 "testparm" 來查看我們的設定有沒有語法上的錯誤。以下為 smb.conf 的主要的設定說明：

```
#================= Global Settings =====================
[global]

# workgroup 就是設定電腦的工作群組。
  workgroup = 企管系

# server string 相當於在 NT 中的電腦描述，就是你的電腦要叫什麼名字
  server string = Samba 伺服器

# 這個設定可以限制連線來源，以增加安全性。我們可以在這裡限制只有
# 本地的機器才可以連線。
;  hosts allow = 192.168.1. 192.168.2. 127.

# 如果希望自動載入列表機清單，而不要一台台設定，可以設為 yes。
  load printers = yes

# 如果你要自己設定 guest account，可以將這裡的註解拿掉，並在
# /etc/passwd 中加入該帳號。如果不設定，預設的帳號是 nobody。
; guest account = pcguest

# Samba 會將每個使用者的使用記錄存成 log.使用者，因此我們在
# /var/log 中建立一個目錄來統一存放這些檔案。
  log file = /var/log/samba/log.%m

# log 檔最大是多少 Kb
  max log size = 50

# 要使用哪一種安全模式。在 Windows 9X 的網路芳鄰中，我們可以設密碼，
```




```
# 而在 Windows NT 中，我們可以設定使用者名稱及密碼。如果在這裡設定為
# share，就是只使用密碼；而設為 user 則是要輸入使用者名稱及密碼。
# 如果我們設為 user，client 端在瀏覽網路芳鄰時，windows 會自動
# 輸入使用者名稱為登入 windows 時所使用的名稱。我們必須要在 samba
# 中加入相對的使用者及密碼。我們等一下會以圖形介面說明如何設定。
  security = user
# 當 security = server 時，可以指定密碼伺服器
;  password server =

# Windows 98 及 WinNT SP3 以上會將密碼加密，我們必須將它設為 yes
 encrypt passwords = yes

# 設定 Samba 可以使用多個介面，如果你有多張網路卡，可以在這裡設定
# 假設你的 ip 是 140.115.25.25，子網路遮罩是 255.255.255.0，你可以設
# 為 140.115.25.25/24
  interfaces = 192.168.1.1/24

# Windows Internet Name Serving Support Section:
;  wins support = yes

# WINS Server - 設定 WINS Server
;  wins server = w.x.y.z

# for Traditional Chinese Users
# 要在網路芳鄰中使用中文必須加入下列設定
client code page=950
; coding system=cap
valid chars = 0xb9
```

```
#================== Share Definitions ===================
# 這個區斷是用來設定我們要分享的資料夾。在這個區斷中，有幾個設定的
# 範例可以讓我們參考。例如，我們要分享的目錄是 /home/share，設定該
# 目錄的分享名稱為 "共享軟體"，只可以讀取不能寫入，而且不必使用密
# 碼，設定為 guest ok=yes 必須要有 security = share 的配合
#
[共享軟體]
  path = /home/share
  guest ok = Yes
  wirteable =no
  browseable = yes

 # 另一個範例，假設我們要一個上傳區，分享路徑為 /home/upload，可以使
 # 用的帳號是 friend，我們必須先用 vipw 建立 friend 的帳號，加入下行
 # friend:*:60000:65534::0:0:Samba user:/home/upload:/sbin/nologin
 # 接著再以指令 smbpasswd -a friend 來建立密碼。然後再修改 upload 目錄
 # 的權限 chown friend /home/upload
 [上傳區]
   path = /home/upload
   username = friend
   read only = No
```

對分享的資料夾除了要在 samba 設定你想要的權限外，對於該目錄在 UNIX 系統上的讀寫權限也要配合。如果你在 Samba 中的設定都正確，卻發現無法對該資料夾寫入，很可能是在系統中的權限沒有正確設定，必須以 chmod 來加以修改。完成了設定之後，我們可以啓動 Samba 了：

```
# /usr/local/etc/rc.d/samba.sh.sample
```

如果要在一開機就啟動 Samba：

```
# cp/usr/local/etc/rc.d/samba.sh.sample /usr/local/etc/rc.d/ samba.sh
```

16.2 使用 swat 設定

除了使用文字編輯軟體來修改 smb.conf 外，我們可以使用 Samba 內建的 swat 來進行設定。首先編輯 /etc/inetd.conf，將最下方 swat 的註解拿掉：

```
swat stream tcp nowait.400 root /usr/local/sbin/swat swat
```

接著再確認 /etc/services 中有沒有下面這一行，如果沒有則自行加入：

```
swat 901/tcp
```

最後重新啟動 inetd：

```
# kill -1 `cat /var/run/inetd.pid`
```

然後就可以使用瀏覽器以 port 901 連到 Samba Server了。假設 Samba 的 ip 是 192.168.0.1，則輸入 http://192.168.0.1:901，被要求輸入帳號密碼時，請輸入 root 及其密碼。接著出現圖16-1的畫面：

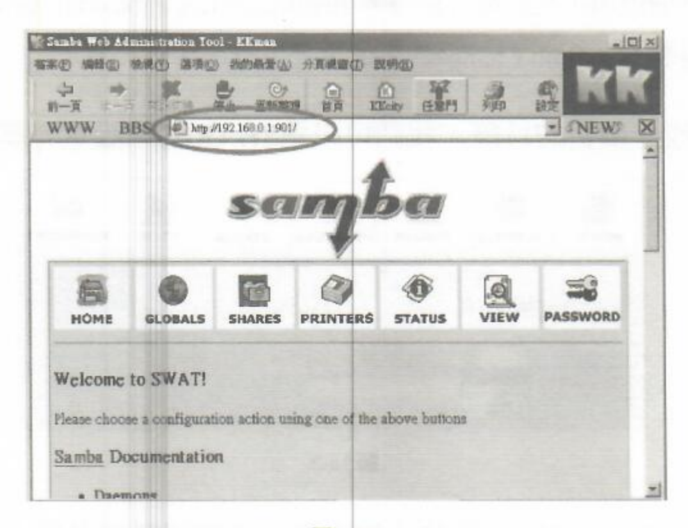


圖 16-1

我們選GLOBALS來看全域設定：

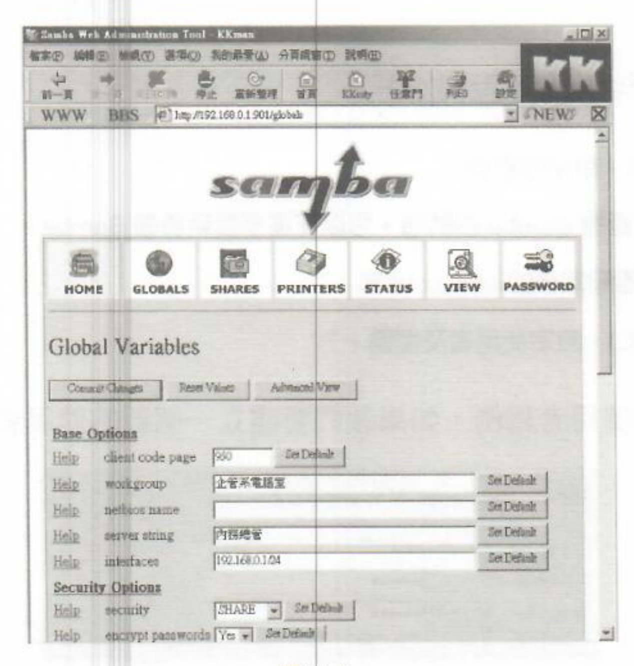


圖 16-2

這裡的每一個選項，我們都可以參考 /usr/local/etc/samba.conf.defaults 來設定，設定完後記得要 " Commit Changes" 來使設定生效。

如果要開放一個新的目錄，我們可以選SHARES來設定分享的資料夾：

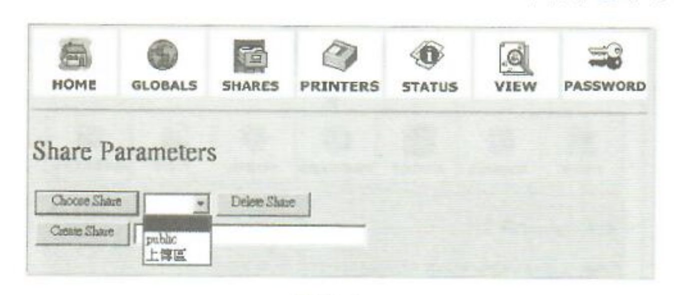


圖 16-3

我們可以在 Create Share 欄位中建立要分享的資料夾名稱，再按 "Create Share" 來設定。或是選擇 "Choose Share"、"Delete Share" 來選擇或刪除已分享的目錄。

其他的各項功能選項說明如下：

- PRINTERS：設定列表機。
- STATUS：查看 Samba 的狀態，可以在這裡重新啓動 Samba。
- VIEW：觀看設定後的 smb.conf 檔。
- PASSWORD：設定使用者及密碼。

以設定新的使用者爲例，如果我們要建立一個新的使用者 friend：

Server Password Management
User Name : friend
New Password : ••••••
Re-type New Password : ••••••
Change Password | Add New User | Delete User | Disable User | Enable User

圖 16-4

使用者 friend 必須已存在於 /etc/passwd 中，否則要自行以 vipw 建立：

```
friend:*:60000:65534::0:0:Samba user:/home/upload:/sbin/nologin
```

建立之後就可以在上圖中 User Name 中輸入 friend 並密碼。這個功能取代了使用 smbpasswd -a 來建立帳號的功能。

如果有任何問題，或是想知道更多關於 Samba 的設定，可以到 Samba 網站(http://www.samba.org/)。

16.3 Windows 設定

MS-windows 系統的設定方面，滑鼠右鍵點選桌面 [網路上的芳鄰] -> [內容]

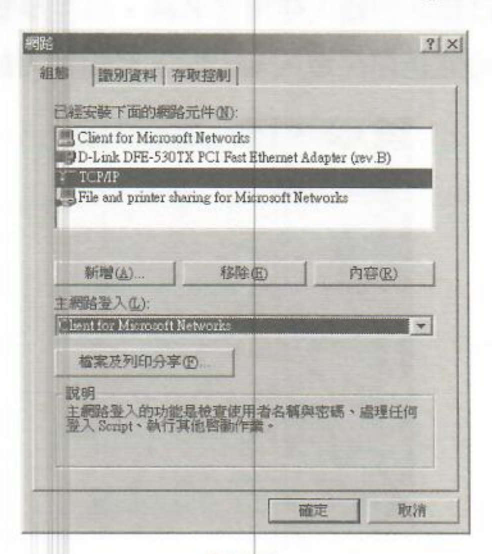


圖 16-5

必須要有 Client for Microsoft Networks 及 File and printer sharing for Microsofe Networks。如果要設定使用者名稱，在主網路登入請選 Client for Microsoft Networks。

工作群組及電腦名稱的設定如下：

網路
組態　識別資料　存取控制
Windows 將依以下資訊，辨識網路上的電腦身份。請輸入電腦名稱、工作群組及電腦說明。
電腦名稱(C): friend
工作群組(O): 企管系電腦室
電腦說明(M): friend
確定　取消

圖 16-6

如果你設定某個目錄必須輸入使用者名稱，但在瀏覽該目錄時卻怎麼輸入密碼都沒有用，可能是你在進入 windows 時所使用的名稱不正確：

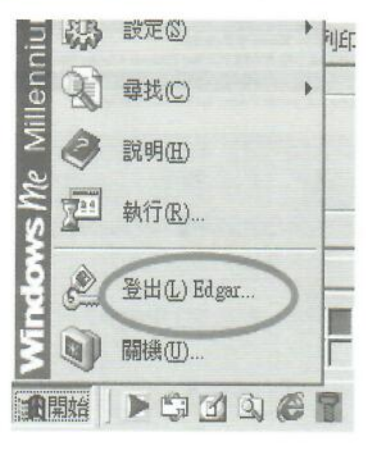


圖 16-7

這時候就必須先登出原本的使用者，再以該目錄要求的使用者名稱登入。

16.4 存取 MS Windows 的網路芳鄰資料

如果您要使用 FreeBSD 來存取 MS Windows 的網路芳鄰資料，FreeBSD 中內建有 mount_smbfs 這個工具，可以讓我們將所要存取的主機及其目錄掛在檔案系統中。掛入後，存取的方式就好像檔案位於硬碟中一樣，十分方便，而且對於中文的存取也沒有問題。mount_smbfs 是在 4.5-RELEASE 之後才內建的，如果您的系統在 4.5-RELEASE 之前，您必須要自行從 ports 中安裝 /usr/ports/net/smbfs。

在使用 mount_smbfs 之前，我們必須先在核心設定中加入下列幾個選項，並重新編譯核心：

```
# SMB/CIFS requester
# NETSMB enables support for SMB protocol, it requires
# LIBMCHAIN and LIBICONV options.
# NETSMBCRYPTO enables support for encrypted passwords.
options     NETSMB          #SMB/CIFS requester
options     NETSMBCRYPTO    #encrypted password support for SMB
# mchain library. It can be either loaded as KLD or compiled into kernel
options     LIBMCHAIN       #mbuf management library
# Kernel side iconv library
options     LIBICONV
options     SMBFS           #SMB/CIFS filesystem
```

編譯完核心並重新啓動後，便可以使用 mount_smbfs 了。

假設我們要存取的主機資料如下：

- IP：192.168.0.2
- 電腦名稱：內務總管
- 要掛入的分享目錄：software
- 掛入系統中哪一個目錄：/mnt

我們可以使用下列指令來掛入：

```
# mount_smbfs -I 192.168.0.2 -N '//內務總管/software' /mnt
```

這裡的參數 I 表示指定 IP 位址，參數 N 表示不須密碼驗證。如果您所要掛入的分享資料夾需要密碼認證，則不要加參數 N。我們在掛入別台電腦的資料夾時，如果沒有指定使用者名稱，內定會以目前所使用的帳號。如果我們要使用別的使用者名稱，可以使用下列方式：

```
# mount_smbfs -I 192.168.0.2 '//username@內務總管/software' /mnt
```

將分享的資料夾掛入後，我們就可以使用 FreeBSD 檔案處理的指令，諸如 cp、mv等來抓取我們所要的檔案，就像是從本機硬碟中使用檔案一樣。

chapter 17 系統安全

17.1 概論

有人說：「沒有不安全的系統，只有懶惰的管理者。」每個系統都有可能會出現漏洞，而有安全性的漏洞產生時，發行的單位都會立即發佈通告及修補的方式，而系統管理者的職責便是要隨時注意是否需要更新漏洞、隨時注意系統是否有異常的訊息。

FreeBSD 相對而言雖然是比較安全的作業系統，但是有時候問題不是在作業系統本身，而是所安裝的軟體。在 FreeBSD 中常見的情形是安裝非 FreeBSD 內定的軟體，但該軟體被收錄在 ports 中，FreeBSD 也會提出警告。因此，我們必須到 FreeBSD 的網站上查看是否有系統安全的消息，網址是 http://www.freebsd.org/security/index.html 。當發現問題時，可以依照發佈的文件中所提供的修正方式來進行修補。

基本上只要電腦電源打開，系統就沒有安全的一天，更何況要連上網路提供服務。系統安全的範圍很廣，從硬體設備的保全、人員管理、網路規劃、到系統本身的管理，我們並不打算說明如何制定一個安全性政策，也無法在這裡說明所有系統安全的相關議題，我們所提及的只是筆者在 FreeBSD 使用上的建議。如果想要更多 FreeBSD Security 的資訊，可以參考 FreeBSD Handbook，我們在安裝 FreeBSD 時已經將 doc 安裝在 /usr/share/doc 中，你可以使用 lynx 來觀看 FreeBSD 的文件。例如：

```
#lynx/usr/share/doc/en_US.ISO8859-1/books/handbook/security.html
```

系統安全並不局限在如何防止他人入侵，對於防止系統內部問題的產生一樣重要。主要的概念就是要讓我們的系統能正常的提供服務，並且對於我們不想讓他人取得的資訊加以保護。然而，爲了系統安全往往必須限制某些功能的使用，而犧牲了便利性。身爲系統管理者往往因爲對於



系統限制太多而受到來自使用者的抱怨，在取捨上本來就不是件容易的事。正因爲如此，一個盡責的管理者在行事上必須具有高度的抗壓性及對安全性的偏執。

17.2 系統管理

17.2.1 執行程式的路徑

有沒有注意到當我們要執行所在目錄中的某一個程式時，例如，在執行所在目錄中的 myscript.sh，我們必須要打 ./myscript.sh。預設的 PATH 中，並沒有將所在目錄 "." 加入路徑中。如果把 "." 加入 PATH 的設定中，可能會產生安全性的問題。如果使用者在 /tmp 中加入一個名爲 ls 的 shell script，內容爲 rm -rf /usr，而我們又將 "." 加入路徑中，當以 root 在 /tmp 中執行 ls 指令時，後果可想而知。因此，我們在執行指令時，最好能指定路徑名稱，如 /bin/ls，並檢查在 shell 設定中是否有將 "." 加入路徑中：

```
# echo $PATH
/sbin:/bin:/usr/sbin:/usr/bin:/usr/games:/usr/local/sbin
```

爲了避免 /bin 及 /sbin 等重要執行檔遭到修改，我們可以爲這些檔案設定禁止修改的 schg flag：

```
# chflags schg /bin/*
# chflags scgh /sbin/*
```

當然，設定了 schg 之後，我們要將 Kernel Security Level 調高到 1 以

上，這樣連 root 都不可以移除 flags。不過設了 schg 之後，我們可能不能執行一些指令，如 make world 等。

17.2.2 降低安裝軟體的風險

我們可以在網路上找到許多免費的軟體，這些軟體固然可以讓我們在系統的使用上更加便利，但卻難保它們不會對系統安全造成任何危害。有的軟體可能存在某些漏洞，即使在我們安裝前尚無任何安全性的問題，日後還是有可能會被人發現軟體的缺陷。因此，我們應該盡可能不要安裝一些雜七雜八的軟體，而安裝之後，一發現有安全性問題也要隨時更新。

不要直接在重要的伺服器上安裝一套新的軟體，最好先在較不重要的電腦上測試，沒問題後再安裝。另外，在安裝軟體時，應注意軟體取得來源是否可靠。如果軟體提供 MD5 或 PGP 的檢查，最好下載後先檢查，再解壓縮。而安裝軟體時，最好取得軟體的原始碼來編譯，我們可以瀏覽程式碼，以了解其架構。閱讀 Makefile 的內容，了解軟體將安裝的確認位置，先確保程式不會在不該出現的地方產生。

17.2.3 kernel Security Level

FreeBSD 中有所謂的 Security Level，它掌控了系統核心的行為運作。只有超級使用者可以使用指令提高 Secruity Level，但不能降低它。如果要降低它必須在 rc.conf 中設定，並重開機。以下為各 Secruity Level 的意義：

- -1：永遠不安全模式。這是預設值，如果設為 -1，它將永遠以 level 0 的模式執行。
- 0：不安全模式。使用者或 root 可以使用 chflags 來移除「不可更動 (immutable)」及 「只能附加 (append-only)」的 flags。所有的裝置只能依其權限來存取。
- 1：安全模式。不可以移除「不可更動 (immutable)」及 「只能附加 (append-only)」的 flags。不可以手動載入或移除 LKM，使用, /dev/mem, and /dev/kmem 只能為唯讀，且不能 newfs 已掛上的檔案系統。
- 2：高度安全模式。除了和安全模式同樣的限制外，不管硬碟是否掛上，都不可以 newfs。另外，kernel time 的改變限制在一秒內，如果超過，會記錄 "Time adjustment clamped to +1 second".
- 3：網路安全模式。除了和安全模式同樣的限制外，還有 IP 封包過濾的規則 (參考 ipfw 及 ipfirewall)，而且不可以調整 dummynet 的設定。

我們可以使用 sysctl 來顯示或設定 Security Level：

```
# sysctl kern.securelevel
```

如果要將 Security Level 設爲 1：

```
# sysctl -w kernel.securelevel=1
```

當我們將 Security Level 設爲 1 以上時，我們會發現沒有辦法安裝新的 kernel (因爲不能移除 schg flag，也沒有辦法使用 big5con 、X Window 等軟體。如果我們的 FreeBSD 只作爲伺服器，而不使用 big5con 或 X Window 的話，可以將 Security Level 的值調高一點。

如果要在開機時設定 Security Level，可以在 /etc/rc.conf 中以下面二行來設定：

```
kern_securelevel_enable="YES" # 是否啟動 Security Level
kern_securelevel="1" # level 從 -1 到 3
```

17.2.4 檢視系統記錄

在 /var/log 中，記錄了許多系統的資訊，我們應該要時常檢視它們。這些檔案如下表：

表27

adduser	使用 adduser 的記錄。
cron	定時排程的記錄。
maillog	郵件記錄。
messages	系統訊息記錄。
security	安全性記錄，如防火牆。

除了系統的記錄外，如果有提供其他服務，會有更多的 log 資料。

如果我們有其他程式為留下 log 檔，最好在 /etc/newsyslog.conf 中設定定時備份壓縮，以免檔案過大。另外，這些備份的 log 檔在 newsyslog.conf 中設定權限 (mode) 時，最好設為 600，以避免其他使用者可以讀取。

FreeBSD 預設每天定時執行一些分析的工作，並將結果寄給 root，建議你最好每天閱讀它們。我們可以在 /etc/mail/aliases 的開頭中加入下面這一行：

```
root: me@my.domain
```

將 my@my.domain 改成你的 email，如此一來，所有寄給 root 的信件，都會自動轉給所設定的信箱。root 每天會收到 "daily run output" 及 "security check output" 這二封信，這是依照我們在 /etc/defaults/periodic.conf 中所設定的定時執行工作輸出的結果。在 daily 執行的任務中，預設並沒有設定定期清除 /tmp，原則上，在開機時系統會清理 /tmp。如果我們不常重開機，可以在 periodic.conf 中設定每天清理 /tmp。

17.2.5 資料的保全

UNIX 系統的安全防護中，第一道防線是電腦實體的安全防護，防止不相干的人接觸電腦及周邊設施。如果很不幸的，外人可接近系統時，第二道防線是系統密碼保護，我們將在下一章說明帳號的防護。然而，如果密碼洩露或被破解，還有第三道防線，就是在 UNIX 系統中的使用者權限及檔案權限控制。如果某一個使用者帳號遭到入侵，我們能限制其活動範圍及資源的存取。而第四道防線就是將重要的資料加以編碼保護，即使資料被使用者竊取，至少還多一道防護措施。而最後一首防線就是資料備份了，我們平時應該有完善的備份計畫，一旦系統發生錯誤或是被摧毀，至少還可以復原。

我們先來談談資料編碼加密的方法，我們可以使用 crypt 這個指令來爲我們的檔案加密。例如，有一個檔名爲 myfile.txt 的檔案，我們使用的金鑰 (key)是 mykey 這個字串，加密後的檔案名爲 myfile.cyp，可以使用下列指令：

```
# crypt mykey < myfile.txt > myfile.cyp
```

加密後，就可以將 myfile.txt 刪除。如果日後要解密，只要執行下列指令：

```
# crypt mykey < myfile.cyp > myfile.out
```

crypt 是一個歷史悠久的編碼軟體，實際上並非十分安全，不過我們可以多加密幾次，讓檔案加密後再加密，只要記得所使用的 key 就好了：

```
# crypt mykey1< myfile.txt | crypt mykey2 | crypt mykey3 > myfile.cyp
```

如果要解密，只要再反過來即可：

```
# crypt mykey3 < myfile.cyp | crypt mykey2 | crypt mykey1 > myfile.out
```

除了 crypt 外，我們也可以使用其他比較好的編碼程式，例如 pgp。pgp 並非 FreeBSD 內附的軟體，但我們可以使用 ports 來安裝它：

```
# cd /usr/ports/security/pgp5
# make install
```

安裝完 pgp 之後，我們必須先產生 key pair。請執行 pgpk -g：

```
$ pgpk -g
Choose the type of your public key:
1) DSS/Diffie-Hellman - New algorithm for 5.0 (default)
2) RSA
Choose 1 or 2: 1 (選擇要使用哪一種方式編碼)

Pick your public/private keypair key size:
(Sizes are Diffie-Hellman/DSS; Read the user's guide for more information)
```

```
1) 768/768 bits- Commercial grade, probably not currently breakable
2) 1024/1024 bits- High commercial grade, secure for many years
3) 2048/1024 bits- "Military" grade, secure for forseeable future(default)
4) 3072/1024 bits- Archival grade, slow, highest security
Choose 1, 2, 3 or 4, or enter desired number of Diffie-Hellman bits
(768 - 4096): 3 (選擇金鑰的長度)
(Producing a 1024 bit DSS and a 2048 bit Diffie-Hellman key)

You need a user ID for your public key. The desired form for this
user ID is your FULL name, followed by your E-mail address enclosed in
<angle brackets>, if you have an E-mail address. For example:
Joe Smith <user@domain.com>
If you violate this standard, you will lose much of the benefits of
PGP 5.0's keyserver and email integration.
(輸入使用者名稱)
Enter a user ID for your public key: John Chung <john@mydomain.com>

Enter the validity period of your key in days from 0 - 999
0 is forever (and the default): 0 (輸入有效期限，0 表示永久)

You need a pass phrase to protect your private key(s).
Your pass phrase can be any sentence or phrase and may have many
words, spaces, punctuation, or any other printable characters.
Enter pass phrase: mykey (輸入一個通行碼，例如 mykey)
Enter again, for confirmation:
Enter pass phrase: (再輸入一次)
Collecting randomness for key...
```

```
We need to generate 541 random bits. This is done by reading
/dev/random. Depending on your system, you may be able
to speed this process by typing on your keyboard and/or moving your mouse.
541 (開始在鍵盤上隨便亂按，來隨機產生)
We need to generate 541 random bits. This is done by reading
/dev/random. Depending on your system, you may be able
to speed this process by typing on your keyboard and/or moving your mouse.
0 * -Enough, thank you.
..******* ........******* .

Keypair created successfully.

If you wish to send this new key to a server, enter the URL of the server,
below. If not, enter nothing. (直接按 Enter)
```

產生了 pgp key 之後，我們就可以使用 pgp 來編碼了。

表28

pgpk -x user -o outfile	取出 user 的公開金鑰至outfile 中
pgpk -a keyfile	將公開鑰匙 keyfile 加入鑰匙環中
pgpk -ll user	印出 user 的公開金鑰
pgpe -r user -at -o outfile txtfile	加密 txtfile 至 outfile
pgps -u use -at -o outfile txtfile	加簽 txtfile 至 outfile
pgpv -o outfile cypfile	解密 cypfile 至 outfile

例如，我們要將 myfile.txt 加密，輸出檔為 myfile.cyp：

```
# pgpe -r john -o myfile.cyp myfile.txt
```

這裡的使用者是 john，他的全名是 John Chung <john@mydomain.com>，我們只要輸入名字的部份關鍵字即可。而解密可以使用：

```
# pgpv -o myfile.out myfile.cyp
```

上面的指令會問你通行碼，輸入後即完成解密。目前 PGP 的主要應用是在於網路傳輸文件的加密，我們可以使用它來為 E-mail 加密，PGP 的用法很多，請自行 man pgp。

不過，即使加密也無法對抗資料的損毀，對於放在電腦中的重要資料，必須維持完整的備份。就算系統都沒有問題，我們永遠不知道哪天會因為我們自己失手刪除重要資料，或是資料遭到破壞。需要備份的不只是程式所產生的資料，自己開發的程式也應該備份。否則即便有了資料可以復原，而沒有程式可以執行。

舉例而言，一個 BBS 站應該要備份的除了使用者資料、精華區及看版文章外，BBS 本身運作的程式也要備份，畢竟那是我們精心撰寫或修改的結晶。備份的資料不應該和運作的系統放在一起，不僅不應放在同一台電腦中，最好不要放在同一個房間、同一棟大樓、同一個城市，甚至同一個國家。

17.3 帳號管理

17.3.1 慎選合宜的密碼

對於系統安全的維護，密碼可以說是最基礎的防線，因此慎選密碼絕對是必要的。不管是超級使用者或是一般的使用者，在設定密碼時，都應該注意一些原則：

- 不要使用和帳號相同的密碼。
- 不要使用字典中找得到的單字，也不要把單字反轉後當成密碼。
- 最好大小寫混用，英文及數字混合，並加入特殊符號。
- 不要使用自己、老婆、小孩的生日、身份証字號等。
- 不要使用鍵盤上連續的字母，如 asdf。
- 不要把密碼寫下來。

一個好的密碼應該是容易記憶，不必另外以紙筆記下的。例如，Gohiy!m (Get out here if you aren't me)，或 ru4@xj4# (以注音輸入法輸入「記錄」)。總之就是要讓別人意想不到，這裡提到的密碼也不要使用。

我們可以使用一些工具來找出系統中密碼太簡單的使用者，例如位於 /usr/ports/security/crack 這套軟體。

17.3.2 控制 root 的使用

在 FreeBSD 中，如果要使用指令 su 來取得 root 的權限，必須將該使用者加入 wheel 群組中。但是 su 並未對使用者執行過的指令留下記錄，如果系統中有許多使用者，我們要針對不同使用者給予不同的權限，也不是 su 所能做到的，因此我們可以使用 sudo。關於 sudo 的使用，請參考 5.5.3 使用者管理一章中「控制 root 的使用」。

root 帳號不應該可以使用 telnet、ssh、或 ftp 的方式登入，在 /etc/ftpuser 中應該有 root 的帳號來限制 root 使用 ftp 登入。而在 /etc/ssh/sshd_config 中，應該有 PermitRootLogin no 的字樣來限制 root 使用 ssh 登入。我們注意的就是避免 root 能從網路上直接登入，以減少安全性的問題。在

/etc/ttys 中，預設了 root 只能有某些 tty 登入系統，這種允許 root 直接登入的 tty 設定中有 secure 的字樣。例如，ttyv0 指的是 console，該行設定最後有 secure，表示 root 可以從該 tty 登入。而 ttyp* 等是遠端登入的 tty，所以禁止 root 直接登入。

17.3.3 限制系統資源的使用

如果使用者登入系統後，執行大量消耗 CPU、記憶體或磁碟的程式，我們的系統將無法正常提供服務，因此，限制使用者對於系統資源的存取是必要的。針對系統所提供的服務，來限制系統的資源，並避免提供不必要的服務。例如，以一台單純的網頁伺服器、DNS 伺服器而言，並不需要開放使用者遠端登入的服務。遠端登入提供使用者對系統有直接的操作，而往往也是漏洞開放的起點。一般使用者對於安全的要求不一定和系統管理者的期望相符，他可能會將密碼寫在自己辦公桌上、或是登入系統執行一些不必要的程式。如果迫不得已一定要爲使用者在系統上開一個帳號，也應該視情況限制其使用遠端登入。如果只要爲使用者開一個 FTP 帳號或是郵件帳號，只要設定使用者所用的 shell 爲 /sbin/nologin 即可。例如，在 /etc/master.passwd 中，使用者 jack 的帳號資料如下：

```
jack:Bk5AI4MiRKDJ4:1000:1000::0:0:Tom Chang:/home/tom:/sbin/nologin
```

就算使用者不能登入，但能使用磁碟空間，所以還是要爲使用者設定磁碟配額。雖說現在硬體價格便宜，但若每個使用者都有數百 MB 的郵件，集合起來也十分驚人。關於磁碟配額的設定，請參考 5.2，使用者管理一章中的「磁碟配額」。在限制使用者郵件容量方面，最簡單的方式就

是將使用者的郵件從 /var/mail 中搬移到使用者的目錄中，再對使用者目錄做磁碟配額的限制。例如，使用者 jack 的家目錄位於 /home/jack，我們可以：

```
# cd /var/mail
# mkdir /home/jack/mail
# mv /var/mail/jack /home/jack/mail/
# ln -s /home/jack/mail/jack
```

如此就可以將使用者原本的郵件放在其家目錄下，我們只要將該檔案再鏈結到 /var/mail 中，如此就可以不必更動郵件軟體的設定，而達到限制空間的效果了。

如果你堅持要讓使用者登入的話，除了磁碟配額外，應該要再為他們設定其他系統資源的使用限制，例名 CPU 的使用量、記憶體等。我們可以經由設定 /etc/login.conf 來做到。而 login.conf 的設定，可以參考安裝設定篇中，/etc 下的檔案介紹。

17.3.4 限制 crontab 及 at 的使用

使用者可以用 crontab 和 at 指令來安排自己定時執行的工作。一般的使用者並不需要擁有 crontab 或 at 的執行權，我們可以為這個指令設限，只允許必要的使用者執行。如果要限制使用 crontab，只需要在 /var/cron 目錄中，加入 allow 或是 deny 這個檔即可。例如，我們只允許少數幾個使用者執行 crontab，我們可以在 /vra/cron 目錄中編輯檔名為 allow 的文字檔，內容為該使用者的名稱。相對的，如果我們要限制少數幾個使用者執行 crontab，只要編輯 deny 這個檔即可。而指令 at 的限制也是一樣，不

同的只是允許執行 at 指令的名單是 /var/at/at.allow，而拒絕的名單是 /var/at/at.deny。

17.4 網路管理

17.4.1 關閉不必要的服務

一台電腦可以提供的服務很多，我們要做的是在許可的範圍內，盡量減少所提供的服務。許多安全性的問題來自於非第三者 (Third Party) 所提供的軟體，如果沒有必要就停止這些服務吧。

我們先來看一下目前系統提供哪些服務：

```
$ netstat -a|grep LISTEN
tcp4  0  0 *.pop3   *.*  LISTEN
tcp6  0  0 *.telnet *.*  LISTEN
tcp4  0  0 *.telnet *.*  LISTEN
tcp6  0  0 *.ftp    *.*  LISTEN
tcp4  0  0 *.ftp    *.*  LISTEN
tcp4  0  0 *.http   *.*  LISTEN
tcp4  0  0 *.https  *.*  LISTEN
tcp4  0  0 *.smtp   *.*  LISTEN
tcp4  0  0 *.ssh    *.*  LISTEN
```

這裡所看到的就是目前系統中所提供的服務。我們可以檢視一下有沒有不必要的服務，並將它移除。最明顯的例子是 sendmail，如果我們的系統

不提供郵件處理，就將它停掉吧。即使我們要提供郵件服務，也應該限制寄信者的身份或來源位址。一台沒有設限的郵件伺服器，最後的結果是惡名昭彰，再也沒有機器會願意轉送我們發送的信件。如果我們要停止 sendmail，只要在 /etc/rc.conf 中加入下面這一行：

```
sendmail_enable="NO"
```

我們可以使用 sysctl 來設定當外部機器要使用我們沒有提供的服務時便記錄下來，例如有人嘗試掃我們的 port，或者我們沒有開放 telnet，卻有人嘗試從 port 23 連接，在 /var/log/messages 中便會留下記錄。這個設定只要執行下列指令：

```
# sysctl -w net.inet.tcp.log_in_vain=1
# sysctl -w net.inet.udp.log_in_vain=1
```

如果要在開機時就啓動這個設定，可以將上面二行指令加到 /etc/rc.local 或是在 /etc/sysctl.conf 中加入下面二行：

```
net.inet.tcp.log_in_vain=1
net.inet.udp.log_in_vain=1
```

另外，FreeBSD 自從 4.4-Release 起，預設將 telnet 及 ftp 的服務也停止了。原因除了 telnet 本身有漏洞外(已修補)，就是這些以明碼方式在網路上傳送使者帳號及密碼的服務其實是系統安全的另一個潛在危險。我們可以檢視一下 /etc/inetd.conf，發現每一行前面都有註解符號 "#"，也就是說目前根本沒有任何經由 inetd 啓動的服務，我們可以經由 /etc/rc.conf 加入下列這一行來停止 inetd 服務：

```
inetd_enable="NO"
```

如果我們必須使用遠端登入來管理系統，不要使用 telnet，請使用 ssh。ssh 對於在網路上流動的資料有加密保護，比起 telnet 安全多了。如果眞的有必要使用 inetd 來啓動某些服務，例如 ftp，建議將使用情形記錄下來。以 ftpd 爲例，FreeBSD 在執行 ftpd 時，內定加上參數 -l，我們要做的只是修改 /etc/syslog.conf。編輯 /etc/syslog.conf 加入下面這一行：

```
ftp.*                    /var/log/ftpd.log
```

這個設定會讓所有登入成功及失敗的記錄都寫在 /var/log/ftpd.log 這個檔案中，我們必需先手動建立 ftpd.log 這個檔案：

```
# touch /var/log/ftpd.log
```

爲了避免 log 檔肥大，我們在 newsyslog.conf 中加入 ftpd.log 的備份移轉：

```
/var/log/ftpd.log     600  5  500  *  Z
```

另外，對於 ftpd 還有一個小建議，爲了避免使用者 ftp 登入後可以到系統所有資料夾，我們最好將使用者的活動範圍限制在自己的家目錄中，這就叫做 chroot。方法很簡單，只要建立一個檔案 /etc/ftpchroot，內容爲使用 chroot 的使用者名稱即可。

其他的 inetd 服務，能不用就不要用，尤其是 telnet。如果要管理電腦，我們可以使用 ssh。

爲了防止一些 DoS (Deny of Service)，建議最好把 ICMP 重導向 (redirect) 的封包丟棄，我們可以在 /etc/rc.conf 中加入以下的設定：

```
icmp_drop_redirect="YES"  # YES 表示丟棄 ICMP REDIRECT 封包
icmp_log_redirect="YES"   # YES 表示將丟棄的封包記錄下來
```

17.4.2 使用ssh

ssh 是一個好用的軟體，FreeBSD 安裝預設啓動 sshd。我們可以檢查一下它是否己啓動：

```
# netstat -a | grep ssh
```

如果沒有，請在 /etc/rc.conf 中加入下面這一行：

```
sshd_enable="YES"
```

sshd 預設並未將使用者登入的資料記錄下來，不過我們可以修改 /etc/syslog.conf 來記錄，請找到 security 的項目，並將它修改成下面這樣：

```
security.*;auth.info                /var/log/security
```

如此一來，當使用者利用 ssh 登入時，便會記錄在 /var/log/security 中。

17.4.3 TCP Wrapper

對於 inetd 所提供的服務，我們可以使用 TCP Wrapper 來限制 TCP 協定連線來源。讓我們來檢視一下 /etc/hosts.allow 這個檔案：

```
ALL : ALL : allow
ftpd : localhost : allow
ftpd : .nice.guy.example.com : allow
ftpd : .evil.cracker.example.com : deny
ftpd : ALL : allow
```

語法：daemon_list : client_list : option

其中 daemon_list 是我們在 /etc/services 中定義的服務名稱，client_list 是來源位址，option 則是我們要給的權限，簡單的設定如 allow(允許)、deny(拒絕)。ALL 可以代表所有服務或來源。這個檔案的設定是以先入爲主 (first match wins) 的方式，也就是以先設定的項目爲優先。

在檔案開頭的地方有一行是 ALL : ALL : allow，表示預設所有服務允許所有來源使用。如果我們要使用 TCP Wrapped，必須先將該行註解，再針對每一個服務來設定開放的權限。以 ftpd 爲例，假設除了 bad.cracker.com 以外，其他人都可以使用 ftpd 服務，我們可以這樣設定：

```
ftpd : bad.cracker.com : deny
ftpd : ALL : allow
```

又如，假設我們的 telnet 只要讓 192.168.0.1 及 mydomain.com 網域下的電腦可以使用：

```
telnetd : 192.168.0.1 .mydomain.com : allow
telnetd : ALL : deny
```

TCP Wrapped 只針對 TCP 服務，如果我們要功能更強大的防火牆，可以使用 ipfw。

17.4.4 ipfw

ipfw 是 FreeBSD 內附的防火牆軟體，它直接針對 IP Layer 來做網路控制，因此可以說是最有效的方法。在使用 ipfw 之前，我們必須先重編核心。關於防火牆的設定，請參考第13章「NAT 及防火牆」。

chapter 18

指令應用

18.1 基本 UNIX 指令

18.1.1 概論

在 UNIX 系統中，使用者對於系統的操作是透過 "Shell"，Shell 就好像是 DOS 中的 command.com 或 Windows 中的 explorer.exe。FreeBSD 內定的 Shell 內是 sh、csh、tcsh，Shell 在接收到指令之後，會將它轉換成機器可以讀的語法來對系統進入操作。

如果我們以 root 登入，所看到的 shell 提示符號為 "#"，如果以一般使用者登入，所看到的提示符號依 shell 的不同會有差異。以 csh 及 tcsh 為例，我們看到的提示符號是 "$"。

在 UNIX 系統中，英文字母的大小寫會被視為不同的東西，因此在輸入指令或檔名時，大小寫的差異要特別注意。一般指定用法的格式大概如下：

```
command [option(s)] [filename(s)]
```

command 是我們要執行的指令。[option] 是我們可以加的參數，用 [ ] 包起來的意思是可以有參數，也可以不加參數。而參數之後，有可能是檔名 [filename]，並不是所有指令都要加參數或檔案名稱，不過格式大部份都是依照這種順序。另外，如果要在命令列中以一行輸入多個指令，每個指令間可以用分號 ";" 分開。

18.1.2 man

查看指令的使用說明。例如我們要看指令 man 的使用說明：

```
$ man man
```

如果我們只知道一個關鍵字，卻不知要使用哪一個指令，我們可以使用參數 k 來查詢。例如我們要查詢 firewall 相關的指令：

```
$ man -k firewall
```

man 在查詢指令說明時，預設會去找 /usr/share/man 目錄下的檔案，如果我們要查詢的指令說明檔並不位於該目錄，我們可以使用參數 M 來指定目錄名稱。例如我們要查詢指定 ab 的用法，該指令的說明檔放在 /usr/local/apache/man ，我們可以使用下列指令：

```
$ man -M /usr/local/apache/man ab
```

我們看 man 檔案時，常會看到像 man(1) 的格式，其中 (1) 表示該指令的分類。依不同的類別，說明檔會存在 /usr/share/man/ 不同的目錄下。例如 (1) 的檔案是在 man1 的目錄中。如果同一個名稱有二個不同的 man file，分別放在不同目錄，我們也可以加上參數來看不同的檔案。例如 crontab 有二個檔案，一個是 crontab(1)，另一個是 crontab(5)。當我們要看 crontab(5) 時，使用下列指令即可：

```
$ man 5 crontab
```

當您使用 man 指令時，所輸出的結果大約如下：

```
NAME
        ls - list directory contents
SYNOPSIS
        ls [-ABCFGHLPRTWabcdfgiklnoqrstu1] [file ...]
DESCRIPTION
        For each operand that names a file of a type other than directory, ls
        displays its name as well as any requested, associated information.  For
        each operand that names a file of type directory, ls displays the        nam
es
        of files contained within that directory, as well as any requested,      asso
-ciated information. .........
EXAMPLES
        The following is how to do an ls listing sorted by size (and shows
why ls does not need a separate option for this):
        ls -l | sort -n +4
        Additionally, the -r flag to sort(1) may be used to get the results
        sorted from largest to smallest (a reverse sort).
SEE ALSO
        chflags(1), chmod(1), sort(1), xterm(1), termcap(5), symlink(7),
        sticky(8)
```

- NAME：指令的簡單描述。
- SYNOPSIS：指令用法，其中 [ ] 所括起來的內容表示該參數可有可無，以 man ls 而言，我們看到 [-ABCF...] 表示可以使用參數 -A -B等。
- DESCRIPTION：指令用法的詳細描述，包括各項參數的使用及限制。
- EXAMPLES：一些用法的範例。
- SEE ALSO：列出其他和本指令相關的指令，我們可以從這些指令中得到更多相關的資訊。



18.1.3 ls

查看目錄資訊。

在 UNIX 系統中，/ 代表根目錄。當要使用某個目錄下的目錄時，每個目錄之間要以 / 隔開。例如 /usr/bin 表示根目錄下的 usr 目錄下的 bin 這個目錄。

另外，"." 和 ".." 也有特殊意義。"." 代表目前所在的目錄，而 ".." 表示目前目錄的上一層目錄。例如，../etc 表示上一層目錄下的 etc 這個目錄。

假設我們要查看根目錄下有哪些檔案：

```
$ ls /
```

在 shell 中，有些符號代表者特殊的意義，例如 * 表示萬用字元，可以代表零個或多個字元，而 ? 代表一個字元。舉例而言，當我們下達 ls 指令來例出檔案時：

```
$ ls myfile*
myfile myfile.exe myfile.txt myfile.txab myfile.abap

$ ls myfile.tx?
myfile.txt
```

我們可以看到使用符號 * 時，會列出所有開頭是 myfile 的檔案；而使用 ? 時，只會例出myfile.txt。

但是當我們使用萬用字元來取代檔案名稱時，例如 ls /m*，它不僅例出符合目錄，還會列出該目錄下所有檔案。這時候我們可以使用參數 d 來讓 ls 只列出目錄而不列出其目錄下的檔案：

```
$ ls -d /m*
```

ls 還有一些常用的參數如下，我們也可以同時使用多個參數，如 ls -lad：

a	列出所有檔案及目錄，包含檔名開頭為 "." 的隱藏檔。
l	列出檔案的完整資訊。
F	依檔案及目錄的格式不同加上符號以供區格，例如目錄則在目錄名稱後加上 / 符號；如果是可執行檔則加上 *；如果是鏈結檔則加上 @。

18.1.4 cd

所在目錄的切換。例如要切換目錄到根目錄：

```
$ cd /
```

切換目錄的方式可以使用絕對路徑或相對路徑名稱。絕對路徑是指從根目錄開始，該目錄所在位置。例如 /usr/bin 就是一個絕對路徑。而相對路徑是指相對於目前所在路徑而言，該目錄的位置。例如 ../usr/bin 表示在上一層目錄下的 usr/bin 這個目錄。另外，符號 "~" 表示使用者的家目錄，如果要回到自己的家目錄，可以使用：

```
$ cd ~
```

如果只輸入 cd 和 cd ~ 所代表的意義相同，都是回到自己的家目錄。我們也可以符號 "~" 之後加上使用者名稱，來代表該使用者的家目錄。例如要切換到使用者 jack 的家目錄：

```
$ cd ~jack
```

18.1.5 pwd

查看目前所在目錄名稱。例如：

```
$ pwd
/root
```

18.1.6 cat

列出文字檔內容。假設我們要查看 /etc/rc.firewall 這個檔案的內容，我們可以使用下列指令來列出：

```
$ cat /etc/rc.firewall
```

在 UNIX 系統中有一個轉向輸出的觀念。我們可以把指令輸出的結果轉向到其他地方 (如檔案)。一般指定的標準輸出是螢幕，標準輸入是鍵盤。我們可以使用 ">" 符號來將輸出轉到別的地方。例如，我們要將 ls 的輸出結果存成檔案 result.txt：

```
$ ls >result.txt
```

上面的指令會建立一個檔名爲 result.txt 的檔案，並將 ls 的結果置於該檔中。如果所在目錄本來就有一個檔案名爲 result.txt，該檔案原本的內容會被清除。如果我們不想清楚該檔原本的內容，只是要把結果附加在原本的內容之後，可以使用 ">>" 。例如：

```
$ ls >> result.txt
```

我們可以使用 cat 指定來做簡單的文字檔複製，例如將 /etc/rc.firewall 複製一份到自己根目錄下的 firewall.txt：

```
$ cat /etc/rc.firewall > ~/firewall.txt
```

另外，我們也可以用 cat 來建立一個文字檔並手動編輯其內容：

```
$ cat >test.txt
```

在此輸入文字
輸入完後同時按 Ctrl+D 離開

18.1.7 more

以分頁模式來列出文字檔內容。當使用 cat 時，如果檔案超過一頁，畫面一閃而過，看都看不清楚。這時候我們可以使用 more 這個指令來將它分頁輸出。

```
$ more /etc/rc.firewall
```

輸出後，我們可以按空白鍵來看下一頁，或按 Q 來離開。

在 UNIX 系統中還有一個觀念是 pipe 管道，就是將一個指令的輸出結果作爲另一個指令的輸入。例如，我們要查看 /etc/ 下的所有檔案完整資訊，使用 ls -la /etc 時，發現資訊超過一頁，我們可以使用下列指令：

```
$ ls -la /etc | more
```



"|" 是位於鍵盤右上角倒退鍵附近，和 "\" 同一個鍵的符號。

18.1.8 less

less 也是以分頁來輸出檔案內容，和 more 不同的是它在輸出檔案全部內容後並不會離開。我們可以使用 page down 及 page up 鍵來查看，要離開時只要按 Q 鍵即可。

18.1.9 head

列出檔案開頭幾行，預設是輸出檔案開頭的十行：

```
$ head /etc/rc.firewall
```

我們也可以加上參數 n 來指定要輸出多少行。例如，如果要輸出前二十行：

```
$ head -n 20 /etc/rc.firewall
```

參數 c 讓我們可以指定要輸出檔案開頭多少 bytes(通常就是多少字元)。例如，如果要輸出檔案開頭前十個字：

```
$ head -c 10 /etc/rc.firewall
```

18.1.10 tail

列出檔案結尾幾行，預設是十行：

```
$ tail /etc/rc.firewall
```

我們一樣可以使用 -n 或 -c 來指定要輸出多少行。

18.1.11 w

列出目前在線上的使用者資訊、時間、正在執行的動作等。

18.1.12 who

列出目前在線上使用者的資訊，輸出的欄位和 w 略有不同。

18.1.13 date

列出及設定系統時間。如果我們要查看目前時間：

```
$ date
```

如果要設定時間為 2002 年 3 月 12 日 11 點 56 分：

```
$ date 200203121156
```

18.1.14 cal

列出月曆。如果要列出當月月曆。

```
$ cal
```

我們也可以指定月份，例如列出 2002 年 3 月。

```
$ cal 3 2002
```

或是列出整年的月曆，例如 2002 年。

```
$ cal 2002
```

18.1.15 echo

輸出一個字串到標準輸出 (通常是螢幕)。例如：

```
$ echo string
```

我們也可以將輸出結果轉向到檔案：

```
$ echo 'this is a test' >test.txt
```

這樣在 test.txt 中就會有一行字串 "this is a test"。

18.1.16 clear

清除螢幕。

18.2 系統管理

18.2.1 ps

在 UNIX 系統中，每個執行中的程式我們稱之爲程序 (Process)，而 ps 這個指令就是用來看目前系統中正在執行的程序狀態。

```
$ ps
PID   TT  STAT  TIME    COMMAND
45836  p0  S     0:00.18 -tcsh (tcsh)
46104  p0  R+    0:00.00 ps
```

PID 欄位指的是 Process ID，就是這個程序的編號，每個程序的編號都是獨一無二的；TT 是指登入的 tty；STAT 是該程序目前的狀態；而 COMMAND 就是這個程序是那一個指令所執行。

ps 不加任何參數時，只輸出自己在執行的程序，我們可以加上參數 -aux 來列出系統中所有使用者的程序及詳細資料。

```
$ps-aux
USER  PID   %CPU %MEM  VSZ   RSS  TT  STAT  STARTED  TIME  COMMAND
root  1     0.0   0.1   552   148  ??  ILs   4        2     0:00.17 int
root  23    0.0   0.0   208   8    ??  IWs   -              0:00.00 adjk
alex  6167  0.0   0.8   1332  988  p0  I     7:46           0:00.21 -csh
root  6241  0.0   0.8   1332  976  p0  D     8:03           0:00.17 _su
```

我們來看看第一行所代表的意義：

- USER：該程序的擁有者。
- PID：Process ID，範圍從 0 到 65535。
- %CPU：該程序目前佔 CPU 使用時間的百分比。
- %MEM：該程序佔用虛擬記憶體的百分比。
- VSM：使用的虛擬記憶體大小。
- RSS：使用的實體記憶體大小。
- TT：登入的 tty。
- STAT：目前的狀態。
- STARTED：開始執行的時間。
- TIME：該程序到目前為止的 CPU 使用時間。
- COMMAND：程序是由哪一個指令執行。

18.2.2 kill

kill 指令會送出一個訊號 (Signal) 給指定的程序，程序在收到訊號之後，會依其要求來動作。可以使用的訊號如下：

表29

SIG名稱	代碼	意　義
HUP	1	hang up，通常用來叫程式重新載入設定。
INT	2	interrup，通知程序中止。
QUIT	3	quit，通知程序離開。
ABRT	6	abort，中斷程序。
KILL	9	non-catchable, non-ignorable kill，直接通知 kernel 把該程序移除。
ALRM	14	alarm clock
TERM	15	software termination signal，通知程序結束。

一般而言，每一個程序都會預設接收到訊號是要執行什麼動作，如果我們所送出的訊號在該程序中並沒特別去處理它，則程序會自動結束程式。只有 root 可以管理所有程序，一般使用者只能對自己的程序作 kill。假設我們要叫 PID 為 123 的程序結束：

```
$ kill -9 123
```

如果要通知程序重新載入設定檔：

```
$ kill -1 123
```

18.2.3 top

top 是一個好用的程序管理程式，我們可以利用它來秀出執行中的程式。進入 top 之後，我們可以按 "h" 來顯示線上說明或按 "q" 來離開。

在 top 中，如果我們想要對某個程序執行 kill 的動作，只要按 "k" 再輸入參數及 PID 即可。

18.2.4 systat

用來監看系統資源使用情形。它有幾個常用的參數：



- -iostat

顯示目前磁碟使用情形，以了解其存取的負荷。

-swap

顯示所有 swap 裝置的使用情形。

-netstat

顯示目前網路連線情形。

進入 systat 之後，我們可以按 Ctrl+L 來重繪畫面，如果要離開 systat 可以先按 : 再打 quit 後離開。

18.2.5 watch

窺視某個 tty 視窗。

當使用者登入系統後，root 可以使用 watch 指令來取得使用者的視窗畫面。也就是說當下達指令後，root 所看到的畫面就會和該使用者一樣。你可以觀察該使用者在做些什麼事，輸出的結果又是什麼。

只有超級使用者 root 可以執行 watch，且在執行前必須先在 kernel 中加入下列的設定並重新編譯核心：

```
pseudo-device snp
```

並新增 snoop device，使用下列指令：

```
# cd /dev
# ./MAKEDEV snp0 snp1 snp2 snp3
```

接下來就可 watch 指令了。首先，先下指令 w 來看一下站上有哪些使用者。指令結果的第二個欄位部份，有使用者的 tty，例如 p0、v0 等，選定要監看的使用者後，使用 watch ttyp0 來監看該使用者，其中 ttyp0 即該使用者的 tty。你可以使用 CTRL+X 來切換不同的 tty，也可以使用 CTRL+G 離開回到自己的畫面。

18.2.6 alias

這是 Shell 內建的指令，用來建立別名。例如，我們希望下達指令 abc 時，會執行 ls /etc，我們可以使用下列指令：

```
$ alias abc 'ls /etc'
$ abc
```

我們可以使用 unalias 來取消所設定的別名，例如：

```
$ unalias abc
```

18.2.7 bg

將指定的程序放入背景中執行。當我們下達一個指令後，必須要等待該程式結束後才能輸入下一個命令。如果該程式必須執行一段很長的時間，我們不想等程式結束，可以把它放到背景中執行。

在下達指令後，按 Ctrl+Z 來暫停程式，接著再執行 bg 即可將程式放到背景中執行。

```
$ sleep 1000
^Z
Suspended
$ bg
[1] sleep 1000 &
$
```

我們也可以在所要執行的指令後面加上符號 "&"，就可以將該程式放到背景執行：

```
$ sleep 1000 &
[2] 46461
```

18.2.8 jobs

jobs 指令可以讓我們查詢目前有哪些程式在背景執行。如果加入參數 -l 可以得到 PID 的資訊。

```
$ jobs -l
[1] + 46459 Running sleep 1000
[2] - 46461 Running sleep 1500
```

18.2.9 fg

將指定的程序放到前景中執行。我們使用 jobs 看到目前在背景執行的程序之後，可以使用 fg 把它叫回前景。例如要叫回第一個在背景中執行的程式：

```
$ fg %1
sleep 1000
```

18.2.10 ntpdate

向時間伺服器對時，只有超級使用者才能執行。我們可以使用 ntpdate 這個指令來向時間與頻率國家標準實驗室的時間伺服器對時：

```
# ntpdate clock.stdtime.gov.tw
12 Mar 21:24:28 ntpdate[46494]: step time server
210.59.157.30 offset -8.939412 sec
```

18.2.11 sync

讓系統暫存的資料強制存回硬碟。

18.2.12 shutdown

讓系統在指定的時間關機。如果我們要立即關機可以下指令：

```
# shutdown now
```

有時候電力公司通知半夜十二點要停電，我們半夜又不想再去使用電腦，這時指定時間關機就發揮作用了。我們可以在白天時先下指令：

```
# shutdown 0203122359
```

0203122359 表示 2002 年 3 月 12 日 23:59，格式是 yymmddhhmm。在指定關機前五分鐘系統會禁止使用者登入，並且會在 /var/run/ 目錄下建立一個檔名爲 nologin 的檔案，內容爲拒絕使用者登入時所要告訴使用者的訊息。如果我們要停止 shutdown ，可以送給它一個 SIGTERM，shutdown 程序在收到訊息後，在離開程式前會先刪除 /var/run/nologin 這個檔案。

```
# ps -ax|grep shutdown
  46644 ?? S<s 0:00.00 shutdown 0203122359
# kill -15 46644
```

shutdown 還有一些參數：

-h	系統停止服務，出現你現在可以放心關機了，但不關閉電源。
-p	系統停止服務並關閉電源。
-r	重新開機。

18.2.13 reboot

立即重新開機。

18.2.14 su

切換使用中的使用者身份。例如，我們要從一般使用者切換成 root：

```
$ su
Password:
#
```

一般使用者如果要具備切換成 root 的權限，必須在 /etc/group 將它加入在 wheel 群組中。例如，我們要該使用者 jack 可以使用 su 變成 root：

```
wheel:*:0:root,jack
```

我們也可以使用 su 來切換成不同的使用者，如果加上參數 -l 表示模擬完全 login 的動作。例如，我們要模擬以使用者 foo 登入系統：

```
# su -l foo
```

我們也可以用參數 -c 來以不同使用者的身份執行一個指令，執行完後切換身份為原本的使用者。例如，我們要以 foo 的身份來執行 sleep 1000 這個指令並放到背景中執行：

```
# su foo -c 'sleep 1000&'
```

18.2.15 exit

這是 Shell 內建的指令，我們可以使用這個指令來登出系統或登出不同的 Shell。

18.2.16 dmesg

顯示系統訊息暫存區 (message buffer) 的內容。如果是剛開機，暫存區的內容通常就是開機過程的記錄。隨著開機時間越來越長，訊息也會越來越多，開機過程的記錄就會被其他訊息所取代。

18.2.17 lastcomm

顯示使用者曾經執行過的指令。如果要使用這個指令，必須在 /etc/rc.conf 中加入下面這一行：

```
accounting_enable="YES"
```

系統會在 /var/account 目錄下建立記錄檔，如果使用者很多的話，檔案大小將會十分可觀。

如果我們要查看使用者 foo 執行過哪些指令：

```
$ lastcomm foo
```

這些記錄每天會自動轉檔，存成 acct.0 acct.1 等檔案。如果我們要查詢

的是前一天的記錄，可以使用參數 -f 來指定使用哪一個記錄檔：

```
$ lastcomm foo -f /var/account/acct.0
```

18.2.18 crontab

安排定時執行工作。使用 crontab 可以讓我們安排工作在指定的年、月、日、小時或分的週期來執行。

如果在 /var/cron 目錄中有 allow 這個檔案的話，只有使用者名稱在檔案中的人才可以使用 crontab。如果沒有 allow 這個檔案，但是有 deny 這個檔案的話，被列在 deny 檔案中的人不可以使用 crontab 來安排工作。如果二個檔案都不存在，預設是所有人都可以執行。

我們可以使用 crontab -e 來編輯自己的排程， 使用 crontab -e 的格式，和 /etc/crontab 的格式不太一樣，它少了執行者的欄位，內定的指令執行者就是執行 crontab -e 的人。其格式如下：

```
MAILTO=""
#分	小時	天	月	星期幾	指令
#minute	hour	mday	month	wday	command
#
*/5	*	*	*	*	setiathome
# minute：代表一小時內的第幾分，範圍 0-59
# hour：代表一天中的第幾小時，範圍 0-23
# mday：代表一個月中的第幾天，範圍 1-31
# month：代表一年中第幾個月，範圍 1-12
# wday：代表星期幾，範圍 0-7 (0及7都是星期天)
```

```
# who：要使用什麼身份執行該指令
# command：所要執行的指令
#
# 時的欄位中如果是 *，表示每小時，天的欄位中如果是 * 表示每天，
# 依此類推欄位中可以使用 "-" 來表示範圍。
# 例如：在小時的欄位中填 8-11，表示執行的時間是8,9,10,11共四次
# 例如：欄位也可以用逗點來表示，以分的欄位而言， 1,2,5,9 表示
#     將在 1,2,5,9 分時各執行一次。也可以寫成像這樣 1-2,12-14
#     表示在 1,2,12,13,14 分各執行一次。
# 又如：以 / 後面加數字表示每幾分鐘要執行一次。如在分的欄位
#     填 0-23/2，表示 1-22 分之間，每隔二分鐘執行一次
#     也就是 0,2,4,6,8,10,12,14,16,18,20,22
# 又如：在分的欄位是 */5，表示每五分鐘一次
#
# 除此之外，也可以用一個開頭為 @ 的字串來表示各種意義
#        字串         代表意義
#        ------       --------
#        @reboot       開機時路一次
#        @yearly       每年跑一次,等於 "0 0 1 1 *".
#        @annually     (和 @yearly 一樣)
#        @monthly      每月跑一次,等於 "0 0 1 * *".
#        @weekly       每週跑一次,等於 "0 0 * * 0".
#        @daily        每天跑一次,等於 "0 0 * * *".
#        @midnight     (和 @daily 一樣)
#        @hourly       每小時跑一次,等於 "0 * * * *".
#
# 安排 crontab 時，應該要錯開每個程式的執行時間，才不會
# 有一大堆程式同時執行。
```

執行 crontab 預設會將指令輸出結果寄 email 給執行的使用者，如果我們不希望收到這些結果，可以在檔案開頭加上 MAILTO=""。

另外，我們也可以使用參數 -l 來列出目前執行的 crontab table。或使用參數 -r 來刪除 table。

18.2.19 uptime

顯示系統開機主機狀況。例如：

```
$ uptime
  10:51下午 up 8 days, 8:46, 2 users, load averages: 1.01, 1.02, 1.00
```

出現的資訊依序爲現在時間、共開機多久、開機時間、目前使用者有多少人、系統每 1 分鐘、每 5 分鐘、每 15 分鐘的平均負荷 (load)。

18.2.20 sysctl

顯示或設定核心 (kernel) 狀態。使用參數 -a 可以列出目前 kernel 狀態值的設定，例如：

```
$ sysctl -a
```

我們也可以使用參數 -w name=value 的方式來設定新的值。這些值如下表：

表30

NAME	TYPE	可否改變
kern.ostype	string	no
kern.osrelease	string	no
kern.osrevision	integer	no
kern.version	string	no
kern.maxvnodes	integer	yes
kern.maxproc	integer	no
kern.maxprocperuid	integer	yes
kern.maxfiles	integer	yes
kern.maxfilesperproc	integer	yes
kern.argmax	integer	no
kern.securelevel	integer	raise only
kern.hostname	string	yes
kern.hostid	integer	yes
kern.clockrate	struct	no
kern.posix1version	integer	no
kern.ngroups	integer	no
kern.job_control	integer	no
kern.saved_ids	integer	no
kern.boottime	struct	no
kern.domainname	string	yes
kern.filedelay	integer	yes
kern.dirdelay	integer	yes
kern.metadelay	integer	yes
kern.osreldate	string	no
kern.bootfile	string	yes
kern.corefile	string	yes
kern.logsigexit	integer	yes
vm.loadavg	struct	no
hw.machine	string	no
hw.model	string	no
hw.ncpu	integer	no
hw.byteorder	integer	no

NAME	TYPE	可否改變
hw.physmem	integer	no
hw.usermem	integer	no
hw.pagesize	integer	no
hw.floatingpoint	integer	no
hw.machine_arch	string	no
machdep.console_device	dev_t	no
machdep.adjkerntz	integer	yes
machdep.disable_rtc_set	integer	yes
user.cs_path	string	no
user.bc_base_max	integer	no
user.bc_dim_max	integer	no
user.bc_scale_max	integer	no
user.bc_string_max	integer	no
user.coll_weights_max	integer	no
user.expr_nest_max	integer	no
user.line_max	integer	no
user.re_dup_max	integer	no
user.posix2_version	integer	no
user.posix2_c_bind	integer	no
user.posix2_c_dev	integer	no
user.posix2_char_term	integer	no
user.posix2_fort_dev	integer	no
user.posix2_fort_run	integer	no
user.posix2_localedef	integer	no
user.posix2_sw_dev	integer	no
user.posix2_upe	integer	no
user.stream_max	integer	no
user.tzname_max	integer	no

假設我們的系統常出現 file: table is full 的訊息，我們可能要重新編譯 kernel 並提高 maxuser 的值。或者我們也可以使用 sysctl 來做更動。首先我們看一下 kern.maxfiles 的值：

```
# sysctl kern.maxfiles
kern.maxfiles: 2024
```

我們可以使用下列指令來提高它：

```
# sysctl -w kern.maxfiles=16244
```

如果我們要讓這個設定在每次重開機時都自動載入，可以將該指令放到 /etc/rc.local 中，或是在 /etc/sysctl.conf 中加入下面這一行：

```
kern.maxfiles=16244
```

18.3 使用者管理

18.3.1 vipw

編修使用者密碼檔。我們可以使用 vipw 這個指令來編輯使用者密碼檔 /etc/master.passwd。如果我們編輯的內容不符合密碼檔的格式，vipw 會提出警告。在修改完後，vipw 還會自動執行 pwd_mkdb 來更新系統資料庫。

18.3.2 groups

這個指令可以秀出使用者屬於哪一個群組。例如秀出使用者 jack 的群組：

```
$ groups jack
wheel  jack
```

18.3.3 adduser

這個指令可以用來新增使用者。執行 adduser 後，它會做以下的動作：

- 在 /etc/group 中加入使用者的群組
- 在 /etc/master.passwd 中加入使用者
- 在 /home 中建立使用者目錄，並建立 dotfile
- 在 /var/mail 中建立使用者郵件目錄
- 送出訊息給使用者

執行指令後，系統會問你一些問題：

```
輸入使用者的shell，我使用比較好用的 tcsh，故輸入 tcsh
Enter your default shell: csh date no sh tcsh [tcsh]:
使用者目錄要放在哪
Enter your default HOME partition: [/home]:
是否要從skel目錄中複製使用者的預設設定檔，直接按 Enter
Copy dotfiles from: /usr/share/skel no [/usr/share/skel]:
是否要送出訊息給使用者，直接按Enter，送出 /etc/adduser.message
Send message from file: /etc/adduser.message no  [/etc/adduser.message]:
是否要使用密碼，按 Enter ，預設是y
Use passwords (y/n) [y]:
使用者的帳號，只能使用小寫的英文字母及數字
Enter username [a-z0-9_-]:
使用者全名，輸入你的真實姓名
Enter full name []:
要使用的 shell，打 tcsh
Enter shell csh date no sh tcsh [tcsh]:
```

```
使用者的目錄，按 Enter 即可
Enter home directory (full path) [/home/asdf]:
使用者的編號，按 Enter 即可
Uid [1001]:
登入的等級，按 Enter 即可
Enter login class: default []:
登入的群組，按 Enter 即可
Login group asdf [asdf]:
是否要該使用者加入其他群組，請注意，因為第一個使用者是你自己，所以在這裡要
打 wheel ，這樣你才可以 su 成 root ，也就是你的帳號才可以登入成 root 帳號
Login group is ``asdf''. Invite asdf into other groups: guest no [no]:
密碼
Enter password []:
再輸入一次
Enter password again []:
以上都正確嗎？
OK? (y/n) [y]:
```

最後還有一些問題，都是直接按 Enter 就可以了，等到它問你是否要再增加下一個使用者時，你可以回答 y 來增加下一個使用者，也可以回答 n 即可回到命令列。

18.3.4 pwd_mkdb

如果我們直接以文書編輯軟體來修改 /etc/master.passwd，在修改完後，必須執行 pwd_mkdb 來將更新的資料做成系統可以讀取的資料庫檔。pwd_mkdb 還會自動建立 /etc/passwd。

18.3.5 rmuser

刪除使用者。使用 rmuser 將會進行下列動作：

- 刪除使用者的 crontab 及 at 指令的資料
- 停止所有該使用者正在執行的程式
- 在密碼檔中刪除使用者
- 如果使用者家目錄的擁有者是該使用者則刪除
- 刪除 /var/mail 中使用者的信件
- 刪除所有在 /tmp, /var/tmp, 及 /var/tmp/vi.recover 的使用者檔案
- 把在 /etc/group 中所有該使用者的名稱刪除。如果使用者所屬群組名稱和使用者名稱一樣，而且該群組是空的，則一並刪除。

必須要注意的是 rmuser 並不能刪除 UID 為 0 的使用者 (如 root)。

18.3.6 passwd

變更使用者密碼。如果不加使用者名稱，則會變更所登入的使用者密碼。用法如下：

```
$ passwd jack
```

18.3.7 chpass

chfn, chsh, chpass 是一樣的東西，用來更改使用者資料，如果以 root 來執行的話，其內容其實就是 master.passwd 的東西。如果以一般使用者執行，使用者可以使用這個指令來修改一些個人資訊。

18.3.8 mesg

是否要允許其他使用者傳送訊息給自己，如果不允許可以使用 mesg n，如果允許則是 mesg y。如果要執行 write 或 talk 必須設為 mesg y。

18.3.9 write

送出訊息給使用者。

用法：write user [tty]

例如，要送出一段訊息給使用者 jack：

```
$ write jack
```

在 jack 的視窗會出現下列訊息：

```
Message from root@foo.com on ttyp0 at 01:39 ...
```

之後所打的每一個訊息都會出現在使用者 jack 的視窗。

18.3.10 last

顯示使用者登入記錄。例如：

```
$ last
mary ttyp0 alex.alexwang 三 3/13 04:01    still logged in
foo   ttyp1 alex.alexwang 三 3/13 03:54 - 04:01 (00:06)
jack  ftp    alex.alexwang 三 3/13 03:53   still logged in
ben   ttyp2 alex.alexwang 三 3/13 03:41 - 03:41 (00:00)
```

如果我們只要顯示某位使用者的登入記錄，我們可以在指令後面加上使用者登入帳號。如果要顯示某一個時間有哪些人在線上，我們可以使用參數 -d。例如，2002 年 3 月 10 日 23:45 時有哪些人在線上：

```
$ last -d 0203102345
```

last 預設會去找 /var/log/wtmp 這個記錄檔，如果你想要使用其他記錄檔可以加上參數 -f：

```
$ last -f /var/log/wtmp.0
```

18.4 檔案系統管理

18.4.1 touch

改變檔案存取的時間。例如：

```
$ ls -l
-rw-r--r-- 1 jack staff 1520505 2/25 20:12 myfile.txt
$ touch myfile.txt
$ ls -l
-rw-r--r-- 1 jack staff 1520505 3/13 15:21 myfile.txt
```

在上面的例子中，原本 myfile.txt 的存取日期是 2/25，我們使用 touch 之後，時間變成目前的時間了。如果使用 touch 時有加檔案名稱，但該檔案不存在，touch 會建立一個空的檔案。

我們也可以使用參數 -t 來指定要將存取時間設定爲什麼時候。例如，我們要設定爲 2002 年 3 月 13 日 23:49：

```
$ touch -t 0203132349 myfile.txt
```

18.4.2 cp

複製檔案或目錄。

用法：cp 來源 目的

我們可以使用 cp 來複製檔案。例如，將 /etc/services 複製到 ~/services.txt：

```
$ cp /etc/services ~/services.txt
```

在上面的指令中，如果我們不指定目地檔名，將 ~/services.txt 改成 ~/ 的話，預設會使用原來的檔名，也就是將 /etc/services 複製到 ~/services。我們也可以同時複製多個檔案到一個目錄下，例如複製 /etc/services 及 /etc/rc.conf 到 ~/：

```
$ cp /etc/services /etc/rc.conf ~/
```

如果要複製目錄，我們可以使用參數 -R 來將來源目錄及其所有子目錄複製到目的地：

```
$ cp -R sourdir tardir
```

18.4.3 ln

建立鏈結檔，所謂的鏈結檔就像在 Windows 下的捷徑。如果我們希望同樣一個檔案或目錄可以放在許多地方，我們可以使用 ln 來建立鏈結檔，這樣一來實際存在的檔案只有一個，但在許多地方都有鏈結。例如我們要將 /etc/services 鏈結到 ~/services：

```
$ ln /etc/services ~/services
```

當我們刪除鏈結檔時並不會影響原本的檔案。鏈結的方式可以分爲 Hard link 及 Symbolic link，預設是使用 Hard link。二者的差別在於 Hard link 只能在相同的檔案系統中建立鏈結，而且不能鏈結目錄。我們在分割

磁碟時，將不同的目錄建立在不同的分割區上，假設 /etc/services 位於 ad0s1a 而使用者的家目錄 ~/ 位於 ad0s1e，那麼上面指定就不會生效。我們必需使用參數 -s 來建立 Symbolic link：

```
$ ln /etc/services ~/services
ln: shit.txt: Cross-device link        <---出現錯誤
$ ln -s /etc/services ~/services
```

我們也可以建立目錄的鏈結：

```
$ ln -s /etc
```

在上面這個指令中，我們沒有指定目的地，預設會在所在目錄建立鏈結。

18.4.4 mkdir

建立目錄。假設我們要在現行目錄下建立一個目錄 temp：

```
$ mkdir temp
```

如果我們要建立一個目錄 /tmp/abc/tmp，在是在 /tmp 下並沒有 abc 這個目錄，我們就必須使用參數 -p 來自動建立：

```
$ mkdir /tmp/abc/tmp
mkdir: /tmp/abc: No such file or directory
$ mkdir -p /tmp/abc/tmp
```

18.4.5 rm

刪除檔案或目錄。例如，我們要刪除 temp 這個檔案：

```
$ rm temp
```

如果要刪除一個目錄，必須使用參數 -r：

```
$ rm -r /tmp/abc
```

在刪除檔案或目錄時，如果該檔案是唯讀的，rm 會詢問使用者是否眞的要刪除，我們可以使用參數 -f 讓 rm 不要詢問直接刪除。或者我們也可以使用參數 -i 來讓 rm 在刪除檔案時不管是否唯讀都要詢問。

18.4.6 mv

搬移檔案或目錄。例如，我們要將 abc 這個檔案搬到 /tmp/test：

```
$ mv abc /tmp/test
```

如果在 /tmp/test 存在，而且是一個目錄的話，那麼 abc 會被放在 /tmp/test/abc。如果 /tmp/test 存在，而且是一個檔案的話，則原來的 test 這個檔案會被刪除，改由 abc 取代之。

我們也可以利用 mv 來更改檔案或目錄名稱。例如，我們要將 abc 改名爲 cde：

```
$ mv abc cde
```

18.4.7 df

顯示磁碟使用情形。

```
$ df
Filesystem    1K-blocks    Used    Avail    Capacity    Mounted    on
/dev/ad0s1a   201518 106212 79186         57%    /
/dev/ad1s1f   2595662      738200 1649810      31%    /home
/dev/ad0s1e   2761230      1615176        925156 64%    /usr
/dev/ad1s1e   503966 11546         452104 2%    /var
procfs 4          4        0                100%   /proc
```

如果你覺得這樣的輸出結果不容易了解，可以使用參數 -h。我們也可以使用參數 -i 來看 i-node 的使用狀況。

18.4.8 du

磁碟使用情形統計。如果我們要看所在目錄使用多少磁碟空間，可以使用 du 這個指令。和 df 指定一樣，我們可以加參數 -h 來顯示較易閱讀的統計格式：

```
$ du -h /etc
```

如果不加目錄名稱 /etc，預設是顯示所在目錄的使用統計。我們可以使用 du 這個指定來看系統中所有使用者的家目錄使用情形：

```
# du -sh /home/*
750M  /home/jack
60M   /home/mary
2M    /home/john
```

如果我們要查出目前系統中使用磁碟空間最大的前5名使用者，我們可以利用 du 搭配 sort 指令：

```
# du -s /home/* | sort -rn | head -5
```

18.4.9 chmod

改變檔案目錄權限。當我們使用 ls -l 時：

```
$ ls -l
drwx------   2 jack  staff     512   2/27 02:14 mail/
drwxr-xr-x   2 jack  staff   77824   2/22 05:37 txts/
-rw-r--r--   1 jack  staff 1520505   3/11 23:39 myfile.txt
drwxr-xr-x  10 jack  staff     512   3/11 05:28 software/
```

第一個欄位代表的是檔案的權限。該欄位中共有十個字元，第一個字元是檔案的類型，其後每三個字元為一組，分別代表使用者 (User)、所屬群組 (Group)、其他人 (Other) 對於該檔案的存取權限。

- r：可以讀取，代表數字 4。
- w：可以寫入，代表數字 2。
- x：可以執行，代表數字 1。

在使用 chmod 時，我們可以使用不同的字母來代表使用者 (User)、所屬群組 (Group)、其他人 (Other)：

- u：User，檔案的擁有者。
- g：Group，擁有者所屬群組。
- o：Other，其他使用者。
- a：All，所有人。

舉例而言，如果我們要讓 myfile.txt 可以讓所有人讀取：

```
$ chmod a+r myfile.txt
```

如果我們要設定和 myfile.txt 擁有者同一個群組的人可以讀取及寫入該檔：

```
$ chmod g=rw myfile.txt
```

如果要設定移除群組對 myfile.txt 寫入的權限：

```
$ chmod g-w myfile.txt
```

另外，我們也可以使用數字來設定檔案權限。r, w, x 都有其對映的數字，以每個使用不同對象爲單位，將所對映的數字相加後所得到的數字就是該對象的權限。

User	Group	Other
rwx 421	rwx 421	rwx 421
rwx 421	r-x 401	r-- 400
↓ 7	↓ 5	↓ 4

圖 18-1

例如，使用者的權限是 rwx，則其權限爲 4+2+1=7。而群組的權限是 rx，其權限爲 4+1=5。其他人的權限是 r，則以數字表示爲 4。我們要設定 myfile.txt 這個檔案的權限：

```
$ chmod 754 myfile.txt
```

我們來看一下關於目錄的權限，目錄的權限中，如果有 x 表示可以進入該目錄，r 表示可以讀取目錄內容，而 w 則是可以對該目錄寫入。我們用下列的例子來說明目錄權限的應用：

```
$ chmod 500 mydir
$ cd mydir
$ ls
file.txt  doc/   mp3/             <--- 權限爲 500，沒問題
$ cd ..
$ chmod 400 mydir
$ cd mydir
mydir: Permission denied. <--- 權限爲 400，只可以讀不能進入
$ ls mydir
file.txt  doc/   mp3/             <--- 權限爲 400，只可以讀不能進入
$ chmod 100 mydir
$ ls
ls: .: Permission denied        <--- 權限爲 100，只能進入，不能看內容
```

我們上面提及的權限都是以三位數字來表示，另外我們也可以使用四位數字表示。所謂的四位數字是指在原本的三位數之前加上一個關於檔案形態的設定。

- 數字 4，set user id (SUID)。表示該檔案在執行時會以檔案擁有人的身份執行。
- 數字 2，如果該檔案可以被執行(具 x 權限)，則在執行時會以擁有者群組的身份執行。如果是不能被執行的檔案，在讀寫時會控制不能讓多個程式同時存取(locked)。
- 數字 1，sticky。如果將檔案設為所有人都可以讀寫，並設定 sticky，則所有人都可以修改該檔案，但是不能刪除。如果是目錄開放讀寫權限，但設定了 sticky，則使用者只能新增檔案，不能刪除，這個可以應用於 FTP 的上傳區。

假設我們要設定檔案 mydir 可以被所有人讀、寫、執行，並設定 sticky：

```
$ chmod 1777 mydir
```

18.4.10 chown

改變檔案的擁有人及群組。例如，我們要將目錄 temp 的擁有人設爲 jack，並設定群組爲 staff：

```
# chown jack.staff temp
```

如果我們要將目錄 temp 及其下所有檔案及子目錄的擁有人改變成 jack，可以使用參數 -R。

18.4.11 chflags

在 FreeBSD 還有一種特別的權限控制，稱之爲「flags」，這些 flags 的設定可以讓我們用來保護特殊的檔案。例如 /kernel 就是一個有設定 flag 的檔案，我們可以使用 ls 加參數 -o 來顯示：

```
# ls -ol /kernel
-r-xr-xr-x 1 root wheel schg 2208222 2/26 02:09 kernel
```

flags 的設定淩駕於一般的權限設定，我們可以設定的主要 flags 及其所代表的意義如下：

表31

nodump	檔案不可以被 dump。(只有檔案擁有者和 root 可以設定)
sappnd	檔案只可以往後附加，不能刪除。(只有 root 可以設定)
schg	檔案不可以被更動，連 root 都不能刪除。(只有 root 可以設定)
uappnd	檔案只可以往後附加，不能刪除。(檔案擁有者和 root 才可設定)
uchg	檔案不可以被更動。(檔案擁有者和 root 才可設定)

如果我們要解除所設定的 flags 只要在上述的 flgs 之前加上 no 即可，例如 nouchg。

用法：chflags flags file

flags 的設定只有在 kernel security level 爲 -1 或 0 時才可以被更改。如果 security level 爲 1 或 2 時就不能更動 flags 了。

18.4.12 umask

當我們新增一個文字檔時，預設的檔案權限是 644，而新增一個可執行檔時，預設的權限是 755，也就是除了檔案的擁有人外，其他人都可以讀取或執行。這種預設權限是由 umask 來控制。

我們看一下 ~/.cshrc 的內容：

```
alias h history 25
alias ls ls -F
# A righteous umask
umask 22
```

這裡的設定是 umask 22，也可以表示爲 umask 022。022 這三個數字分別代表擁有者、群組、其他人的權限，建立檔案是，將檔案所有權限減去這些數字後，所得到的值就是檔案的預設權限。

例如，一般檔案權限的全部權限是 666，分別減去 022 後，得到的預設權限就是 644。而可執行檔的權限是 777，分別減去 022 後，就是 755。知道了 umask 之後，或許我們會希望將它設爲 077。當然，設爲 077 可能會產生一些問題，例如建立一個網頁後，可能沒有辦法讓它在使用者的瀏覽器中出現 (因爲執行 apache 通常是以 nobody 的身份來執行)。

我們可以使用 umask 來顯示目前的設定，或使用 umask num 來設定 umask：

```
$ umask
22
$ umask 77
$ umask
77
```

18.4.13 diff

比較二個檔案的差異。例如：

```
$ diff file1 file2
```

18.4.14 wc

計算行數(lines)、字數(words)、位元數(bytes)。我們可以使用這個指令來計算檔案中的字數：

```
$ wc file.txt
  77 103 1076 file.txt
```

輸出結果分別代表行數、字數、位元數，我們也可以使用參數 -l, -w, -c 來指定要輸的是行數、字數、位元數。我們可以將 wc 和 ls 一起使用來計算檔案數：

```
$ ls /etc | wc -l
100
```

UNIX 指令的好處就在於我們可以自行組合指令創造出新的用法。

18.4.15 whereis

找尋程式的所在。whereis 預設會去尋找標準的二進位檔、說明檔、及原始程式碼檔名符合的檔案。所以我們不能用它來找一般文字檔。

```
$ whereis netstat
netstat: /usr/bin/netstat /usr/share/man/man1/netstat.1.gz
```

18.4.16 which

在使用者的路徑設定中尋找該程式。

```
$ which perl
/usr/bin/perl
```

18.4.17 find

在指定目錄下尋找檔案。find 可以用的參數很多：

表32

-name file	尋找檔名為 file 的檔案。
-perm mode	尋找權限為 mode 的檔案。
-size n[c]	尋找檔案大小為 n block 的檔案。c 表示字元數。
-atime n	尋找在 n 天之前曾被存取的檔案。
-mtime n	尋找在 n 天之前曾被更改時間的檔案。
-ctime n	尋找在 n 天之前曾被更改內容的檔案。
-newer file	尋找修改時間比 file 新的檔案。
-print	找到之後，列出檔名。
-exec cmd {} \;	找到之後執行 cmd 指令，在 cmd 最後一定要加上 \; 指令才會執行。如果 cmd 後有加 {} 表示執行的目錄在該檔案的目錄。
-user name	尋找擁有者為 name 的檔案。
-group name	尋找群組為 name 的檔案。
-nouser	尋找使用者名稱不在 /etc/passwd 中的檔案。
-nogroup	尋找群組不在 /etc/group 中的檔案。

例如，我們要從根目錄開始，找出 services 這個檔案所在位置：

```
$ find / -name service -print
```

如果我們要找出檔案大小大於 10M 的檔案：

```
$ find / -size +10485760c -print
```

如果我們要從所在目錄開始，找出所有副檔名爲 bak 的檔案，並將它刪除：

```
$ find . -name '*.bak' -exec rm {} \;
```

18.4.18 grep

找尋某一個字串。例如，我們要找 /etc/ 下所有檔案內容有 192.168 這個字串的檔案：

```
$ grep '192.168' /etc/*
```

我們也可以將 grep 和其他指令一起使用，例如要找出現在執行的程序中 inetd 的 PID：

```
$ ps -aux | grep inetd
```

18.4.19 tar

Tape archiver。可以用來壓縮備份檔案。tar 的用法很多，我們僅介紹簡單的壓縮與解壓縮。舊版的 tar 並不具有壓縮功能，只是把檔案包裝成一個磁帶檔。現在的 tar 都可以加上參數 -z 來順便將檔案壓縮。

表33

參數	用途
z	壓縮檔案。
x	取出檔案。
c	建立檔案。
f file	指定要處理的檔案。
v	觀看過程。
u	更新檔案，新的檔案會取代較舊的檔案。
r	將檔案附加在原本的 tar 檔之後。

例如，我們要將目錄 temp 包裝並壓縮成 temp.tgz 這個檔案：

```
$ tar zcvf temp.tgz temp
```

要解開 temp.tgz：

```
$ tar zxvf temp.tgz
```

18.4.20 fsck

檢查並修復檔案系統。我們可以指定要修復的檔案系統，或不加任何參數來檢查所有檔案系統。fsck 當檢查發現有問題時，預設會詢問使用者是否要修復，我們也可以加參數 -y 來對於所有問題都回答 y。

```
# fsck /dev/ad0s1e
```

18.4.21 mount

掛入檔案系統。如果要掛入的檔案系統在 /etc/fstab 中有記錄，則可以不必指定來源：

```
# mount /usr
```

表34

-a	掛入所在在 /etc/fstab 檔中記錄的檔案系統，有參數 noauto 者除外。
-o options	設定檔案系統參數。options 參數如下： async，非同步寫入模式。 noexec，該檔案系統上的檔案不可以被執行。 nosuid，該檔案系統不允許 set uid 或 set gid 的檔案發生作用。 nosymfollow，在該檔案系統上不可使用鏈結。 rdonly，該檔案系統是唯讀的，連 root 也不可以寫入。 sync，使用同步寫入模式。
-t type	設定要掛入的檔案系統格式。如果要掛入的格式不是內定的格式，mount 會去呼叫 /sbin/mount_XXX 的程式來使用。例如要掛入 msdos 系統時，會去使用 /sbin/mount_msdos 這個程式。常用的格式如下： ufs，本機的 UNIX 檔案格式。 nfs，Network File System。 msdos，DOS 檔案格式。 isofs，CD-ROM (ISO-9660) 檔案格式。

例如我們要掛入一個 MS-DOS 的磁片到 /mnt 的目錄中：

```
$ mount -t msdos /dev/fd0 /mnt
```

也可以使用 mount_msdos 指令：

```
# muont_msdos /dev/fd0 /mnt
```

我們在使用 CD-ROM 之前要先將它掛入，如果是使用光碟安裝 FreeBSD，在 /etc/fstab 中有 CD-ROM 的設定，我們只要執行下列指令：

```
# mount /cdrom
```

如果沒有，我們要知道光碟機的代號，可以使用 dmesg 來查看開機記錄檔中關於 CD-ROM 的訊息。假設我們的光碟代號是 cd0c，要將它掛入 /cdrom：

```
# mount -t  cd9660 /dev/cd0c /cdrom
```

或是：

```
# mount_cd9660 /dev/cd0c /cdrom
```

18.4.22 unmount

移除掛入的檔案系統。只要輸入 unmount mount_point 即可。

CD-ROM 在掛入之後，每法取出光碟片。我們必須使用 unmount 來移除才可以將光碟退出：

```
# unmount /cdrom
```

18.5 網路相關指令

18.5.1 ping

檢查遠端系統的連線狀態。ping 指令會送出 ICMP 封包到指定的主機，我們可以藉此來檢查網路連線品質。

常用參數如下：

- -c count　　指定要計算 count 次。
- -s size　　指定每個封包大小為 size。
- -t timeout　　指定 time out 時間。
- -I interface　　如果目標主機位址是廣播位址，而且我們有多個網介面，可以指定要使用哪一個介面。

例如我們要看 www.freebsd.org 的連線品質：

```
$ ping www.freebsd.org
PING freefall.freebsd.org (216.136.204.21): 56 data bytes
64 bytes from 216.136.204.21: icmp_seq=0 ttl=54 time=458.986 ms
64 bytes from 216.136.204.21: icmp_seq=1 ttl=54 time=502.258 ms
64 bytes from 216.136.204.21: icmp_seq=2 ttl=54 time=491.489 ms
^C
--- freefall.freebsd.org ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 458.986/484.244/502.258/18.393 ms
```

18.5.2 ifconfig

設定或檢查網路介面。我們可以使用 ifconfig 來顯示所有的網路介面，如果使用參數 -u 表示顯示使用中的網路介面，而 -d 則是非運作中的介面。

我們也可以使用 ifconfig 來讓網路介面運作或停用。

用法：ifconfig [down|up] interface

假設我們要讓網路卡 vr0 停用：

```
# ifconfig vr0 down
```

我們也可以使用 ifconfig 來設定網路上的 IP 位址。假設要設定 IP 為 192.168.0.1，而子網路遮罩為 255.255.255.0：

```
# ifconfig vr0 192.168.0.1 netmask 255.255.255.0
```

接著再使用 ifconfig 將 vr0 啟用：

```
# ifconfig vr0 up
```

18.5.3 arp

顯示 arp 位址。例如我們要顯示 192.168.0.2 這台機器的網路卡號：

```
$ arp 192.168.0.1
```



18.5.4 traceroute

追蹤由本機到某台主機所使用的路徑。當我們使用 ping 來檢查網路連線狀況時，如果發現無法連線，我們可以使用 traceroute 來檢查到底是網路上的哪一台主機有問題。

```
$ traceroute www.freebsd.org
```

18.5.5 netstat

顯示網路狀況。我們可以使用 netstat 來顯示目前的連線狀況。例如：

```
$ netstat -a
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address Foreign Address   (state)
tcp4      0      20     www.ssh   198.z27z4z49.1780 ESTABLISHED
tcp4      0      0       *.http          *.*            LISTEN
tcp4      0      0       *.https         *.*            LISTEN
tcp4      0      0       *.smtp          *.*            LISTEN
tcp4      0      0       *.ssh           *.*            LISTEN
tcp4      0      0       *.pop3          *.*            LISTEN
udp4      0      0       *.syslog        *.*
udp6      0      0       *.syslog        *.*
Active UNIX domain sockets
Address  Type   Recv-Q Send-Q    Inode    Conn Refs Nextref Addr
cd864e00dgram      0       0       0         cd84ef00 cd864fc0
cd864fc0dgram      0       0       0         cd84ef00       0
```

我們可以由上面的結果看到目前有一個使用者正使用 ssh 連到我們的網站。經由上表，我們可以看出我們所提供的服務有哪些，目前的使用情形如何。如果我們希望 Foreign Address 直接顯示 IP ，可以使用參數 -n。

我們也可以使用參數 -i 來查看網路介面的使用情形：

```
$ netstat -ai
Name  Mtu   Network       Address            Ipkts Ierrs  Opkts Oerrs  Coll
dc0   1500  <Link#1>     00:80:c8:f6:b2:66 68890922 15997 8370716  1256 60296
                  33:33:c0:f6:78:e9
dc0   1500  fe80:1::280 fe80:1::281:c8ff:        0   -      0   -    -
                  ff02:1::2:c1f7:78e9(refs: 1)
                  ff02:1::1       (refs: 1)
                  ff02:1::1:ffe7:b266(refs: 1)
lp0*  1500  <Link#2>                             0    0      0    0    0
lo0   16384 <Link#3>                         34050    0  34050    0    0
```

我們簡單說明一下各欄位所代表的意義：

- Name：設備的名稱。
- Mtu：最大的傳送單元(unit)。
- Network：此介面所提供的網路或目的地主機。
- Address：介面的位址。
- Ipkts：表示接收到的封包。
- Ierrs：表示接收到但破損的封包數量。
- Opkts：表示送出的封包。
- Oerrs：表示送出但破損的封包。
- Coll：表示發生碰撞 (Collision) 次數。當網路負荷量大時，封包送出時較易發生碰

撞，碰撞產生時，系統會等待一段時間嘗試再次送出封包。碰撞次數越多，連線品質越差。

18.5.6 sockstat

列出開啓中的 socket。

```
$ sockstat
USER    COMMAND   PID  FD PROTO  LOCAL ADDRESS  FOREIGN ADDRESS
root    telnetd 52897   0 tcp4   192.168.0.1:23 192.168.0.2:1969
root    sshd    34063   4 tcp4   *:22           *:*
nobody  httpd   11670  16 tcp4   *:443          *:*
nobody  httpd   11670  17 tcp4   *:80           *:*
root    sendmail  117   4 tcp4   *:25           *:*
root    sendmail  117   5 tcp4   *:587          *:*
root    inetd     109   4 tcp4   *:21           *:*
USER    COMMAND   PID  FD PROTO  ADDRESS
mysql   mysqld    170   6 stream /tmp/mysql.sock
root    sendmail  117   3 dgram  syslogd[100]:3
root    syslogd   100   3 dgram  /var/run/log
```

每個欄位所代表的意義如下：

- USER：哪個使用者開啓的 socket。
- COMMAND：經由哪一個指令。
- PID：該指令的 process ID 是多少。
- DF：socket 的 file descriptor number。

- PROTO：哪一種協定。
- LOCAL ADDRESS：本地的位址及 port (Internet sockets only)。
- FOREIGN ADDRESS：來源的位址及 port (Internet sockets only)。
- ADDRESS：socket 開啓的檔案或目的程式(UNIX sockets only)。

18.5.7 mail

郵件處理程式。古老的 UNIX 郵件處理程式，這個程式對於不熟悉的人使用起來可能有點困難。但是這是在每個 UNIX 系統中都會有的程式，有時在沒有其他選擇的狀況下，我們還是要使用它，至少要知道如何用它來收發信件。

假設我們要寄信給本機的 root：

```
$ mail root
```

如果我們要寄信給非本機的使用者，可以使用的收件人格式如下：

```
user@cc.ncu.edu.tw
user@\[140.115.1.13\]
```

執行了 mail 之後，程式會先要求我們輸入郵件主旨，輸入後就可以開始打本文了。當完成本文的編輯之後，可以按 Ctrl+D 來將信件送出，或是按二次 Ctrl+c 取消。

我們也可以在執行 mail 時加上參數 -s "subject" 來指定主旨：

```
$ mail -s "hi, my friend" jack@mymail.com
```

如果我們想要將一個文字檔的內容當做本文送出，例如，我們可以先編輯一個文字檔 content.txt，接著使用下列方式：

```
$ mail -s "hi, my friend" jack@mymail.com < content.txt
```

說完了寄信，我們來了解一下如何收信。我們可以打 mail 來收信，如果是 root 還可以使用 mail -u user 來收使用者 user 的信件。

```
$ mail
"/var/mail/root": 12 messages 10 unread
>U  1 jack        Fri Feb 22 03:02  42/690   "Hi friend"
 U  2 jack        Fri Feb 22 03:02  74/2620  "see you tomorrow"
 U  3 mary@abc.com Sat Feb 23 03:06 570/33527 "don't forget"
&
```

最後面出現的 & 爲 mail 程式命令列的提示符號，在第一封信件開頭有一個符號 ">" 表示目前作用中的信件。我們可以直接輸入郵件編號來讀取信件。另外，我們也可以輸入下列指令：

表35

h	列出所有信件。
r	回覆目前作用中的信件。
n	讀取下一封信件。
p	讀取前一封信件。
pre [mail number]	保留編號為 mail number 的信件在系統的 mailbox中 (/var/mail/)。程式 mail 對於已讀取的信件，預設會將它搬到使用 者家目錄下的 mbox 中。我們如果日後還想要使用 pop3 來收該信件，就必須使用 pre 將該信件保留在系統的郵件目錄中。
d	刪除作用中的信件。
z	顯示下一頁信件列表。
q	離開 mail。

18.5.8 telnet

使用終端機遠端登入網路上的主機。例如：

```
$ telnet bbs.mgt.ncu.edu.tw
```

如果在 telnet 時要能輸入中文，必須加上參數 -8：

```
$ telnet -8 bbs.mgt.ncu.edu.tw
```

18.5.9 ssh

使用 telnet 並未加資料加密，我們很容易在不知不覺中洩露資訊。如果要登入的主機有提供 ssh 登入的話，最好使用 ssh。

用法：ssh username@hostname

ssh hostname

例如：

```
$ ssh mary@140.115.77.11
$ ssh jack@mydomain.com
$ ssh mydomain.com
```

只打 hostname 而沒有使用者名稱，登入名稱會是你目前所用的使用者名稱。

如果所連線的主機是第一次連線會出現下列一堆東西，打 "yes" 三個字即可：

```
The authenticity of host '140.115.77.11' can't be established.
RSA key fingerprint is 13:93:8a:61:31:df:41:3f:7a:0a:77:ad:7e:49:e7:3f.
Are you sure you want to continue connecting (yes/no)? yes
```

18.5.10 ftp

檔案傳輸程式。如果要登入的主機允許匿名登入，我們使用參數 -a 來自動登入。

```
$ ftp -a freebsd.csie.nctu.edu.tw
```

進入 ftp 之後，會出現命令的提示列。我們可以輸入以下的指令：

表36

help 或 ?	顯示可以使用的指令。
ls	列出遠端所在目錄的檔案。
pwd	顯示遠端所目錄位置。
cd dir	進入遠端的 dir 目錄。
get file	從遠端取回 file 檔案。
put file	將本地端的 file 檔案上傳到遠端機器。
acsii	使用文字模式傳送檔案。
binary	使用二進位模式傳送檔案。
bye	結束 ftp。
mget *.tgz	取回遠端所有名稱為 *.tgz 的檔案。
mput *.tgz	上傳本地所有 *.tgz 的檔案。
!ls	顯示本地所在目錄下的檔案。
lpwd 或 !pwd	顯示本地所在目錄。
lcd [dir]	切換本地所在目錄。

18.5.11 nslookup

網路主機名稱查詢。如果我們要查詢 www.freebsd.org 所對映的 IP，最簡單的用法是：

```
$ nslookup www.freebsd.org
```

我們也可以在上述指令最後面加上要查詢的 DNS 主機：

```
$ nslookup www.freebsd.org dns.hinet.net
```

我們也可以使用 IP 來進入反查：

```
$ nslookup 216.136.204.21
```

18.5.12 dig

是另一個功能強大的主機名稱查詢工具。簡單的用法如下：

```
$ dig -x 216.136.204.21
$ dig www.freebsd.org
```

18.5.13 tcpdump

顯示或記錄網路封包。如果要使用 tcpdump，在核心中必須要有 Berkeley packet filter，而且有 /dev/bpf*。如果沒有請在核心設定中加入下面這一行，並重新編輯核心：

pseudo-device bpf

執行 tcpdump 後，它會打開指定介面的 promiscuous mode (介面必須支援才有用)。所謂的 promiscuous mode(雜亂模式) 是指不管是否和本機有關的封包都接收進來，要達到這樣的效果，必須藉由 bpf 的支援。

我們可以使用 tcpdump 來觀察到達某一個網路介面的封包。例如我們要監看介面 vr0 的封包：

```
# tcpdump -i vr0
```

如果要結束直接按 Ctrl+C 即可。 如果限制封包數量，可以使用參數 -c。我們也可以使用參數 -w 來將捕捉到的封包存成檔案，在這裡我們存成 dump 這個檔案：

```
# tcpdump -c 20 -i vr0 -w dump
```

使用參數 -r 可以讀取儲存的封包資料：

```
# tcpdump -r dump
```

為了控制 tcpdump 能 dump 我們想要的封包，我們還可以在指令最後加上一些 expression 來控制封包的記錄。關於 expression 的用法請 man tcpdump。

另外介紹一個好用的分析工具 tcpshow， 我們可以使用 ports 來安裝。

```
# cd /usr/ports/net/tcpshow
# make install clean
```

接著就可以使用 tcpshow 來分析我們儲存的封包內容：

```
# tcpshow <dump | more
-----------------------------------------------------------------
Packet 8
TIME:   04:53:10.938750 (0.011744)
LINK:   00:80:2D:BB:65:38 -> 00:50:AA:00:DC:DD type=IP
  IP:   tw -> 189 hlen=20 TOS=00 dgramlen=44 id=4353
        MF/DF=0/1 frag=0 TTL=52 proto=TCP cksum=C56B
 TCP:   port http -> 2451 seq=3298970558 ack=2899053999
        hlen=24 (data=0) UAPRSF=010010 wnd=65535 cksum=8549 urg=0
DATA:   <No data>
-----------------------------------------------------------------
```

我們簡單的說明一下這個封包的內容。第一部份是時間 TIME。

第二行是 LINK，顯示了來源 -> 目的地的網路卡號，另外經由 type=IP，我們知道這是一個 Ethernet_II 的 frame。

第三部份 IP，tw -> 189 是來源及目的地的位址。hlen 是 header length 大小是 20 bytes，而整個 IP 封包 (dgramlen) 的大小是 44 bytes。

第四部份是 TCP，來源是的 port 是 http (內定是 80)，而目的地的 port 是 2451。接下來是 TCP 封包的 sequence number 及 acknowledgement 編號。TCP 的 header length 是 24，加上 IP 的 header 20 長度剛好是 44，和 dgramlen 的長度一樣，這個封包應該沒有破損。

最後，這個封包並沒有包含其他的資料。

chapter 19

Shell Script

19.1 概論

Shell Script 是一個類似 MS Windows 中 .bat 檔的東西，簡單的說，Shell Script 就是將一堆 shell 中的指令放在一個文字檔中來執行。因此，爲了能寫出一個 shell Script，你必須先對 UNIX 指令有初步的認識。身爲一個 UNIX 系統的管理者，一定要會使用 shell script 來使管理工作更加容易。

一般我們會將 Shell Script 的副檔名命名爲 .sh，但並非一定要這麼做，這樣做只是爲了要更容易管理這些檔案。在介紹如何 Shell Script 的內容之前，先來看如何寫出一個 Shell Script 並執行它。假設要寫一個名爲 test.sh 的 Shell Script，首先用你習慣使用的文字編輯軟體來開一個檔案名爲 test.sh 內容如下：

```
#!/bin/sh
echo Hello world!!
```

第一行是必需的，用來定義你要使用的 shell。這裡我們定義要使用的是 Bourne Shell，其所在路徑是 /bin/sh。在 UNIX 系統中有許多不同的 Shell 可以使用，而每個 Shell 的特性及用法都有些許的不同。因此，在寫 Shell Script 時，我們會針對 Bourne Shell (sh) 來寫，因爲 sh 是所有 UNIX 系統中都會有的 Shell。就算你執行 Shell Script 時的環境不是使用 sh，只要加上第一行 #!/bin/sh 就可以在執行此 Shell Script 時使用 sh。而第二行的 echo 代表列出一個字串，我們常使用它來輸出資訊。將 test.sh 存檔後，我們就可以用下列其中一種方式執行它：

1. 轉向輸入

```
sh < test.sh
```

2. 如果要輸入參數的話，第一種方式便不適用，可以改用這種方法。

<arguments> 就是我們要輸入的參數，在上面的 test.sh 中並不需要輸入參數：

```
sh test.sh <arguments>
```

3.你也可以改變 test.sh 的權限，將它變成可以獨立執行的檔案，這樣就可以只打 test.sh 來執行它：

```
chmod a+x test.sh
./test.sh
```

在 Shell Script 中，你們可以使用 # 爲註解，在 # 後面的字串都將被視爲註解而被式忽略。而分號 ; 則代表新的一行，例如打 ls;ls -d 代表二個指令。另外，我們可以使用變數、流程控制、甚至是副函式來使程式更加靈活。以下的各章節我們會詳細加以說明。

19.2 變數的使用

19.2.1 變數的使用

我們知道 Shell Script 是使用一堆指令拼湊而成，爲了方便說明及練習起見，我們不使用編輯檔案的方式來執行，而改以在命令列中打我們要的指令。首先，先打 sh 來進入 Bourne Shell。

```
sh
$
```

在打了 sh 之後，會進入 Bourne Shell，其一般使用者的提示字元爲 $。以下各指令開頭的 $ 表示提示字元，而 $ 之後的字串才是我們輸入的字串。

在 Shell Script 中，所有的變數都視爲字串，因此並不需要在定義變數前先定義變數類型。在 Shell 中定義和使用變數時有些許的差異。例如，我們定義一個變數 color 並令它的值爲 red，接著使用 echo 來印出變數 color 的值：

```
$ color=red
$ echo $color
red
```

在這裡，以 color=red 來定義變數 color 的值爲 red，並以 echo $color 來印出 color 這一個變數。在定義變數時，不必加 $，但是在使用它時，必須加上 $。請注意，在等號的二邊不可以有空白，否則將出現錯誤 ，系統會誤以爲你要執行一個指令。

我們再介紹一個範例：

```
$ docpath=/home/td/src/doc
$ echo $docpath
/home/td/src/doc
$ ls $docpath
abc.txt    abc2.txt    semmt.doc
$ ls $docpaht/*.txt
abc.txt    abc2.txt
```

這裡我們定義了變數 docpath 的值爲 /home/td/src/doc，並印出它。接著我們使用 ls 這個指令來印出變數 docpath 目錄中所有檔案。再以 ls $docpath/*.txt 來印出 /home/td/src/doc/ 目錄下所有副檔名爲 .txt 的檔案。

我們再來看一個例子，說明如何使用變數來定義變數：

```
$ tmppath=/tmp
$ tmpfile=$tmppath/abc.txt
$ echo $tmpfile
/tmp/abc.txt
```

另外，我們也可以使用指令輸出成爲變數，請注意這裡使用的二個 ` 是位於鍵盤左上角的 ，在 shell script 中，使用包起來的代表執行該指令：

```
$ now=`date`
$ echo $now
Mon Jan  14  09:30:14 CST  2002
```

如果在變數之後有其他字串時，要使用下列方式來使用變數：

```
$ light=dark
$ echo ${light}blue
darkblue
$ echo "$light"blue
darkblue
```

這裡雙引號中的字將會被程式解讀，如果是使用單引號將直接印出 $light 而非 dark。

經由上面幾個簡單的例子，相信您對變數的使用已有初步的認識。另外有一些我們必須注意的事情：

```
$ color=blue
$ echo $color
blue
$ echo "$color"
```

```
blue
$ echo '$color'
$color
$ echo \$color
$color
$ echo   one    two    three
one two three
$ echo "one    two    three"
one    two    three
```

我們可以看到上面各個執行結果不大相同。在 Shell Script 中，雙引號 " 內容中的特殊字元不會被忽略，而單引號中的所有特殊字元將被忽略。另外，\ 之後的一個字元將被視爲普通字串。

19.2.2 程式會自動定義的變數

在執行 Shell Script 時，程式會自動產生一些變數：

表37

$?	表示上一個指令的離開狀況，一般指令正常離開會傳回 0。不正常離開則會傳回 1、2 等數值。
$$	這一個 shell 的 process ID number
$!	最後一個在背景執行的程式的 process number
$-	這個參數包含了傳遞給 shell 旗標 (flag)。
$1	代表第一個參數，$2 則為第二個參數，依此類推。而 $0 為這個 shell script 的檔名。
$#	執行時，給這個 Shell Script 參數的個數
$*	包含所有輸入的參數，$@ 即代表 $1, $2,....直到所有參數結束。$* 將所有參數無間隔的連在一起，存成一個單一的參數。也就是說 $* 代表了 "$1 $2 $3..."。

$@	包含所有輸入的參數，$@ 即代表 $1, $2,....直到所有參數結束。$@ 用將所有參 數以空白為間隔，存在 $@ 中。也就是說 $@ 代表了 "$1" "$2" "$3"....。

以下我們舉幾個例子來說明：

```
$ ls -d /home
/home
$ echo $?
0
$ ls /home/aaa/bb/ccc
/home/aaa/bb/cc: No such file or directory
$ echo $?
2
$ echo $?
0
```

上面例子中的第一行是 ls，我們可以看到存在一個目錄 /home，接者 echo $? 時，出現 0 表示上一次的命令正常結束。接著我們 ls 一個不存在的目錄，再看 $? 這個變數變成 2，表示上一次執行離開的結果不正常。最後一個 echo $? 所得到的結果是 0，因爲上一次執行 echo 正常顯示 2。

如果寫一個檔案名爲 abc.sh，內容如下：

```
#!/bin/sh
echo $#: $1 $2 $3 $4 $5 $6 $7 $8 $9
echo $@
```

接著以下列指令來執行該檔案：

```
$ chmod a+x abc.sh
$ ./abc.sh a "b c d" e f
4:a b c d e f
a b c d e f
```

上面最後二行即為執行結果。我們可以看到 $# 即為參數的個數，而 $1, $2, $3...分別代表了輸入的參數 "a", "b c d", "e", "f"，而最後的 $@ 則是所有參數。

19.2.3 系統內定的標準變數

你可以使用 set 這個指令來看目前系統中內定了哪些參數。一般而言會有 $HOME, $SHELL, $USER, $PATH 等。

```
$ echo $HOME
/home/jack
$ echo $PATH
/usr/bin:/usr/sbin:/bin
```

19.2.4 空變數的處理

如果程式執行時，有一個變數的值尚未被給定，你可以利用下列方式來設定對於這種情形提出警告：

```
$ echo $number one
one
```

```
$ set -u
$ echo $number one
sh: ERROR: number: Parameter not set
```

在 set -u 之後，如果變數尚未設定，則會提出警告。你也可以利用下列的方式來處理一些空變數及變數的代換：

表38

${var:-word}	如果變數 var 尚未設定或是 null，則將使用 word 這個值，但不改變 var 變數的內容。
${var:=word}	如果變數 var 尚未設定或是 null，則變數 var 的內容將等於 word 這個字串，並使用這個新的值。
${var:?word}	如果變數 var 已經設定了，而且不是 null，則將使用變數 var。否則則印出 word 這個字串，並強制離開程式。我們可以設定一個字串 "Parameter null or not set" 來在變數未設定時印出，並終止程式。
${var:+word}	如果變數 var 已經設定了，而且不是 null，則以 word 這個字串取代它，否則就不取代。

我們以下面的例子來說明：

```
$ echo $name Wang
Wang
$ echo ${name:-Jack} Wang
Jack Wang
$ echo $name Wang
Wang
```

上面的例子中，變數 $name 並未被取代，而下面的例子中，$name 將被取代：

```
$ echo $name Wang
Wang
$ echo ${name:=Jack} Wang
```

```
Jack Wang
$ echo $name Wang
Jack Wang
```

19.3 運算符號

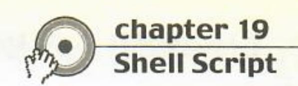
19.3.1 四則運算

在 shell 中的四則運算必須使用 expr 這個指令來輔助。因爲這是一個指令，所以如果要將結果指定給變數，必須使用 ` 包起來。請注意，在 + - * / 的二邊都有空白，如果沒有空白將產生錯誤：

```
$ expr 5 -2
3
$ sum=`expr 5 + 10`
$ echo $sum
15
$ sum=`expr $sum / 3`
$ echo $sum
5
```

還有一個要特別注意的是乘號 * 在用 expr 運算時，不可只寫 *。因爲 * 有其他意義，所以要使用 * 來代表。另外，也可以用 % 來求餘數。

```
$ count=`expr 5 \* 3`
$ echo $count
$ echo `expr $count % 3`
5
```



我們再列出更多使用 expr 指令的方式，下列表中爲可以放在指令 expr 之後的表達示。有的符號有特殊意義，必須以 \ 將它的特殊意義去除，例如 *，否則必須用單引號將它括起來，如 '*'：

表39

類別	語法	說明
條件判斷	expr1 | expr2	如果 expr1 不是零或 null 則傳回 expr1，否則傳回 expr2。
	expr1 \& expr2	如果 expr1 及 expr2 都不為零或 null，則傳回 expr1，否則傳回 0。
四則運算	expr1 + expr2	傳回 expr1 加 expr2 後的值。
	expr1 - expr2	傳回 expr1 減 expr2 後的值。
	expr1* expr2	傳回 expr1 乘 expr2 後的值。
	expr1 / expr2	傳回 expr1 除 expr2 後的值。
	expr1 % expr2	傳回 expr1 除 expr2 的餘數。
大小判斷	expr1 \> expr2	如果 expr1 大於 expr2 則傳回 1，否則傳回 0。如果 expr1 及 expr2 都是數字，則是以數字大小判斷，否則是以文字判斷。以下皆同。
	expr1 \< expr2	如果 expr1 小於 expr2 則傳回 1，否則傳回 0。
	expr1 = expr2	如果 expr1 等於 expr2 則傳回 1，否則傳回 0。
	expr1 != expr2	如果 expr1 不等於 expr2 則傳回 1，否則傳回 0。
	expr1 \>= expr2	如果 expr1 大於或等於 expr2 則傳回 1，否則傳回 0。
	expr1 \<= expr2	如果 expr1 小於或等於 expr2 則傳回 1，否則傳回 0。
文字處理	expr1 : expr2	比較一固定字串，即 regular expression。可以使用下列字元來輔助： . 匹配一個字元。 $ 找字串的結尾。 [list] 找符合 list 中的任何字串。 * 找尋 0 個或一個以上在 * 之前的字。 \(\) 傳回括號中所匹配的字串。

我們針對比較複雜的文字處理部份再加以舉例：

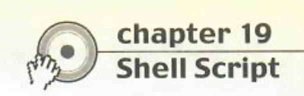
```
$ tty
ttyp0
$ expr `tty` : ".*\(..\)\$"
p0
$ expr `tty` : '.*\(..\)$'
p0
```

上面執行 tty 的結果是 ttyp0，而在 expr 中，在 : 右側的運算式中，先找 .* 表示0個或一個以上任何字元，傳回之後在結尾 ($) 時的二個字元 \(..\)。在第一個 expr 的式子中，因為使用雙引號，所以在 $ 之前要用一個 \ 來去除 $ 的特殊意義，而第二個 expr 是使用單引號，在單引號內的字都失去了特殊意義，所以在 $ 之前不必加 \。

19.3.2 簡單的條件判斷

最簡單的條件判斷是以 && 及 || 這二個符號來表示。

```
$ ls /home && echo found
found
$ ls /dev/aaaa && echo found
ls: /dev/aaaa: No such file or directory
$ ls -d /home || echo not found
/home
$ ls /dev/aaaa && echo not found
ls: /dev/aaaa: No such file or directory
```



a && b	如果 a 是真，則執行 b。如果 a 是假，則不執行 b。
a || b	如果 a 是假，則執行 b。如果 a 是真，則不執行 b。

19.3.3 以 test 來比較字串及數字

我們說過 Shell Script 是一堆指令的組合，所以在比較字串及數字時一樣是經由系統指令來達成。這裡我們使用 test 及 [來做運算，運算所傳回的結果是眞 (true) 或假 (false)。我們可以將它應用在條件判斷上。test 和 [都是一個指令，我們可以使用 test 並在其後加上下表中的參數來判斷眞假。或者也可以使用 [表達示] 來替代 test，要注意的是 [ ] 中的空白間隔。

-n str1	如果字串 str1 的長度大於 0 則傳回 true。
-z str1	如果字串 str1 的長度等於 0 則傳回 true。
str1	如果字串 str1 不是 null 則傳回 true。
str1 = str2	如果 str1 等於 str2 則傳回 true。等號二邊有空白。
str1 != str2	如果 str1 不等於 str2 則傳回 true。!= 的二邊有空白。
a -eq b	Equal，等於。a 等於 b 則傳回真 (true)。
a -ne b	Not equal，不等於。a 不等於 b 則傳回真 (true)。
a -gt b	Grwater than，大於。a 大於 b 則傳回真 (true)。
a -lt b	Less Than，小於。a 小於 b 則傳回真 (true)。
a -ge b	Greater or equal，大於或等於。a 大於或等於 b 則傳回真 (true)。
a -le b	Less or equal，小於或等於。a 小於或等於 b 則傳回真 (true)。

我們舉例來說明：

```
$ test 5 -eq 5 && echo true
true
$ test abc!=cde && echo true
ture
```

```
$ [ 6 -lt 10 ] && echo true
ture


$ pwd
/home
$ echo $HOME
/home/jack
$ [ $HOME = `pwd` ] || echo Not home now
Not home now
```

19.3.4 以 test 來處理檔案

我們也可以使用 test 及 [來判斷一個檔案的類型。下表中爲其參數：

表40

-d file	如果 file 為目錄則傳回真(true)。
-f file	如果 file 是一般的檔案則傳回真(true)。
-L file	如果 file 是連結檔則傳回真(true)。
-b file	如果 file 是區塊特別檔則傳回真(true)。
-c file	如果 file 是字元特別檔則傳回真(true)。
-u file	如果file 的 SUID 己設定則傳回真(true)。
-g file	如果file 的 SGID 己設定則傳回真(true)。
-k file	如果file 的 sticky bit 己設定則傳回真(true)。
-s file	如果 file 的檔案長度大於 0 則傳回真(true)。
-r file	如果 file 可以讀則傳回真(true)。
-w file	如果 file 可以寫則傳回真(true)。
-x file	如果 file 可以執行則傳回真(true)。

我們舉例來說明：

```
$ [ -d /bin ] && echo /bin is a directory
/bin is a directory
$ test -r /etc/motd && echo /etc/motd is readable
/etc/motd is readable
```

第一個指令測試 /bin 是否存在，而且是一個目錄，如果是則執行 echo 傳回一個字串。第二個指令是測試 /etc/motd 是否可以被讀取，如果是則執行 echo 傳回一個字串。

19.4 內建指令

在 Shell 中有一些內建的指令，這些內建的指令如流程控制及 cd 等指令是 Shell 中的必備元素。另外還有一些爲了提高執行效率的指令，如 test、echo 等。有的內建指令在系統中也有同樣名稱不同版本的相同指令，但是如 test、echo 等在執行時會僞裝成是在 /bin 中的指令。

在寫 shell script 時，要注意指令是否存在

下列即爲常見的內建指令：

表41

exit	離開程式，如果在 exit 之後有加上數字，表示傳回值，如：exit 0。在 UNIX 系統下，當程式正常結束，會傳回一個值 0，如果不正常結束則會傳回一個非 0 的數字。
. file	dot 指令，在 shell 中可以使用 "." 來呼叫一個外部檔案，例如 . /etc/rc.conf 或 . .profile。注意 . 和其後的指令中間有空白。
echo	印出一個字串。如果要使用非 shell 內建的 echo 則打 /bin/echo 來使用。
pwd	顯示目前所在目錄。
read var ...	從標準輸入 (通常是鍵盤) 讀入一行，然後將第一個字指派給跟在 read 之後的第一個參數，第二個字給第二個參數，依此類推，直到最後將所有字給最後一

	個參數。如果只有一個參數則將整行都給第一個參數。
readonly [var..]	readonly 這個指令如果沒有加參數則顯示目前唯讀的變數。如果有加變數的話，則將該變數設定為唯讀。
return [n]	離開所在函式，如果在其後有加數字的話，則傳回該數字。和 exit 一樣，這個指令可以傳回該函式的執行結果，0 表示正常結束。

set	將 $1 到 $n 設定為其參數的字。例如：
	
	$ date
	Mon Jan 21 11:19 CST 2002
	$ set \`date\`
	$ echo $4
	11:19

wait [n]	等待在執行程序 (PID) 為 n 的背景程式結束，如果沒有加參數 n 則等待所有背景程式結束。
exec command	執行一個外部程式，通常用於要改變到另一個 shell 或是執行不同的使用者介面，如： exec /usr/local/bin/startkde
export [var]	設定環境變數，如果沒有參數則印出新的環境變數。
eval command	把參數當成 shell 命令來執行，如： $ a=c; b=m; c=d; cmd=date $ eval $\`echo $a$b$c\` Mon Jan 21 11:19 CST 2002

19.5 流程控制

19.5.1 if 的條件判斷

基本語法：

```
if condition-list
    then list
elif condition-list
    then list
else list
fi
```

範例一：

```
#!/bin/sh
if test -r /etc/motd
    then cat /etc/motd
else  echo "There is not motd or file is not readable"
fi
```

說明：上面這一個程式是檢查 /etc/motd 這個檔案是否可以讀，如果可以則印出該檔案，否則印出檔案不可讀。

範例二：

```
$ cat test.sh
#!/bin/sh
if [ $1 -gt 5 ]
```

```
    then echo " $1 is bigger then 5"
elif [ $1 -ge 0 ]
    then echo " $1 is between 5 and 0. "
else echo "$1 is less then 0."
fi

$ ./test.sh 3
3 is between 5 and 0.
```

說明：這裡我們建立一個檔名為 test.sh 的檔案，以指令 cat test.sh 來看它的內容。接著執行 ./test.sh 3，表示輸入一個參數 3。test.sh 檔案的內容表示依輸入的參數判斷參數大於 5 或介於 5 和 0 的中間，或者是小於 0。

19.5.2 while 及 until 迴圈

基本語法：

```
while condition-list
  do list
  done
until condition-list
  do list
  done
```

範例一：

```
#!/bin/sh
i=1
```

```
while [ $i -le 5 ]
  do
    echo $i
    i=`expr $i + 1`
  done
```

說明：首先令變數 i=1，接著在迴圈中當 i 小於等於 5 時就印出 i 的值，每印一次 i 就加 1。直到 i 大於 5 才停止。

範例二：

```
#!/bin/sh
i=1
until [ $i -gt 5 ]
  do
    echo $i
    i=`expr $i + 1`
  done
```

說明：首先令變數 i=1，接著迴圈會判斷，一直執行到 i 大於 5 才停止。每跑一次迴圈就印出 i 的值，每印一次 i 就加 1。注意 while 和 until 的判斷式中，一個是 -le ，一個是 -gt。

19.5.3 for 迴圈

基本語法：

```
for name in word1 word2 ...
  do do-list
  done

for name
  do do-list
   done
```

範例：

```
$ cat color.sh
#!/bin/sh
for color in blue red green
   do
      echo $color
   done

$ ./color.sh
blue
red
green
```

說明：這個檔案 color.sh 中，會在每一次迴圈中將關鍵字 in 後面的字串分配給變數 color，然後印出變數 color。關鍵字 in 讓我們可以依序設定一些值並指派給變數，然而，我們也可以不使用關鍵字 in。如果沒有關鍵

字 in ，程式會自動讀取輸入的參數，並依序指派給 for 之後的變數。請看範例二。

範例二：

```
$ cat color1.sh
#!/bin/sh
for color
  do
    echo $color
  done

$ ./color1.sh  black green yellow
black
green
yellow
```

說明：在 color1.sh 這個檔中，for 迴圈沒有使用 in 這個關鍵字。但我們在執行它時輸入三個參數，迴圈會自動將輸入的參數指派給 for 之後的變數 color，並印出它。

19.5.4 case 判斷

基本語法：

```
case word in
  pattern1) list1 ;;
  pattern2) list2 ;;
```




```
...
esac
```

範例：

```
$ cat num.sh
for num
do
  case $num in
    0|1|2|3)      echo $num is between 0~3;;
    4|5|6|7)      echo $num is between 4~7;;
    8|9)          echo $num is 8 or 9;;
    *)            echo $num is not on my list;;
  esac
done

$ ./num.sh 3 8 a
3 is between 0~3
8 is 8 or 9
a is not on my list
```

說明：這個程式是用來判斷輸入的參數大小。for 迴圈會將每一個輸入的參數指定給變數 num，而在 case 中，判斷變數 num 的內容符合哪一個條件，同一個條件中的每個字用 | 分開。如果未符上面的條件則一定會符合最後一個條件 * 。每一個要執行的 list 是以 ;; 做結尾，如果有多行 list，只要在最後一行加上一個 ;; 即可。

19.6 函式的運用

在 Shell Script 中也可以使用函式 (function) 來使用程式模組化。

基本語法：

```
name ( )
{
   statement
}
```

函式有幾個要注意的地方：

- 在使用函式之前一定要先定義它，也就是在一個 Shell Script 中，一定要先寫函式的內容，在檔案最後再寫會呼叫函式的程式部份。
- 在 Shell Script 中的變數全部都是全域變數 (Global)，所以在函式中的變數也會影響函式外的其他部份。
- 命令列輸入的參數在 Shell Script 中是以 $1, $2....來讀取，但是這些參數並不會在函式中出現。所以必須使用傳遞參數的方式來將要在函式中使用的變數傳給該函式。傳遞的方法和在命令列中使用 Shell Script 的方式一樣，例如：name arg1 arg2..。傳進函式的變數會以 $1,$2... 來儲存，這和命令列傳給 Shell Script 的參數名稱一樣但內容不同。

範例：

```
$ cat test.sh
#! /bin/sh
ERRLOG=$1
```

```
ok ( )
{
    read ans
    case $ans in
        [yY]*) return 0;;
        *) return 1;;
    esac
}
errexit ( )
{
    echo $1
    date >> $ERRLOG
    echo $1 >> $ERRLOG
    exit
}
echo -n  "Test errexit function [y/n] "
ok && errexit "Testing the errexit function"
echo Normal termination

$ ./test.sh err.log
```

說明：

這個程式中有二個函式：errexit 及 ok。第一行定義要將 log 檔存在傳給這個 Shell Script 的第一個參數。接著是二個函式，之後印出一行字，echo -n 表示印出字後游標不換行。然後再執行 ok 這個函式，如果 ok 函式執行成功則再執行 errexit 函式，並傳給 errexit 函式一個字串，最後再印出一個字串。

在 ok 函式中，使用 read 指令來讀入一個參數並指派給變數 ans。接著判斷使用者輸入的值是否為 Y 或 y，如果是則傳回 1 代表沒有成功執行，如果不是則傳回 0 代表成功執行函式 ok。

如果 ok 函式傳回 1 便不會執行 errexit 函式。如果是 0 則在 errexit 函式中，會先印出要傳給 errexit 的參數 " Testing the errexit function"，並記錄在指定的檔案中。

附錄 A 版權宣告

A.1 The FreeBSD Copyright

Copyright 1994-2002 FreeBSD, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE FREEBSD PROJECT ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FREEBSD PROJECT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the FreeBSD Project or FreeBSD, Inc.

A.2 The 4.4 BSD Copyright

All of the documentation and software included in the 4.4BSD and 4.4BSD-Lite Releases is copyrighted by The Regents of the University of California.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:
   This product includes software developed by the University of California, Berkeley and its contributors.
4. Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUEN-

TIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Institute of Electrical and Electronics Engineers and the American National Standards Committee X3, on Information Processing Systems have given us permission to reprint portions of their documentation.

In the following statement, the phrase ``this text'' refers to portions of the system documentation.

Portions of this text are reprinted and reproduced in electronic form in the second BSD Networking Software Release, from IEEE Std 1003.1-1988, IEEE Standard Portable Operating System Interface for Computer Environments (POSIX), copyright C 1988 by the Institute of Electrical and Electronics Engineers, Inc. In the event of any discrepancy between these versions and the original IEEE Standard, the original IEEE Standard is the referee document.

In the following statement, the phrase ``This material'' refers to portions of the system documentation.

This material is reproduced with permission from American National Standards Committee X3, on Information Processing Systems. Computer and Business Equipment Manufacturers Association (CBEMA), 311 First St., NW, Suite 500, Washington, DC 20001-2178. The developmental work of Programming Language C was completed by the X3J11 Technical Committee.

The views and conclusions contained in the software and documentation are those of the authors and should not be interpreted as representing official policies, either expressed or implied, of the Regents of the University of California.

A.3 GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING,
DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a

"work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus form ing a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part con tains or is derived from the Program or any part thereof, to be licensed as a whole at

no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:



a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corre sponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.
If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under

this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is

intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose dis tribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING

RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

> *one line to give the program's name and an idea of what it does.*
> *Copyright (C) yyyy name of author*
>
> *This program is free software; you can redistribute it and/or*
> *modify it under the terms of the GNU General Public License*
> *as published by the Free Software Foundation; either version 2*
> *of the License, or (at your option) any later version.*
>
> *This program is distributed in the hope that it will be useful,*
> *but WITHOUT ANY WARRANTY; without even the implied war-*

ranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details
type `show w'. This is free software, and you are welcome
to redistribute it under certain conditions; type `show c'
for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright
interest in the program `Gnomovision'
(which makes passes at compilers) written
by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

A.4 GNU LIBRARY GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the library GPL. It is
numbered 2 because it goes with version 2 of the ordinary GPL.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Library General Public License, applies to some specially designated Free Software Foundation software, and to any other libraries whose authors decide to use it. You can use it for your libraries, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library, or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link a program with the library, you must provide complete object files to the recipients so that they can relink them with the library, after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

Our method of protecting your rights has two steps: (1) copyright the library, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the library.

Also, for each distributor's protection, we want to make certain that everyone understands that there is no warranty for this free library. If the library is modified by someone else and passed on, we want its recipients to know that what they have is not the original version, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that companies distributing free software will individually obtain patent licenses, thus in effect transforming the program into proprietary software. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License, which was designed for utility programs. This license, the GNU Library General Public License, applies to certain designated libraries. This license is quite different from the ordinary one; be sure to read it in full, and don't assume that anything in it is the same as in the ordinary license.

The reason we have a separate public license for some libraries is that they blur the distinction we usually make between modifying or adding to a program and simply using it. Linking a program with a library, without changing the library, is in some sense simply using the library, and is analogous to running a utility program or application program. However, in a textual and legal sense, the linked executable is a combined work, a derivative of the original library, and the ordinary General Public License treats it as such.

Because of this blurred distinction, using the ordinary General Public License for libraries did not effectively promote software sharing, because most developers did not use the libraries. We concluded that weaker conditions might promote sharing better.

However, unrestricted linking of non-free programs would deprive the users of those programs of all benefit from the free status of the libraries themselves. This Library General Public License is intended to permit developers of non-free programs to use free libraries, while preserving your freedom as a user of such programs to change the free libraries that are incorporated in them. (We have not seen how to achieve this as regards changes in header files, but we have achieved it as regards changes in the actual functions of the Library.) The hope is that this will lead to faster development of

free libraries.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, while the latter only works together with the library.

Note that it is possible for a library to be covered by the ordinary General Public License rather than by this special one.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Library General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus form ing a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties

under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be sup plied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful. (For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the

notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this

License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also compile or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must

be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

c) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

d) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and

distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among coun-

tries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Library General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose dis tribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER

EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

one line to give the library's name and an idea of what it does.
Copyright (C) year name of author

This library is free software; you can redistribute it and/or
modify it under the terms of the GNU Library General Public
License as published by the Free Software Foundation; either
version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Library General Public License for more details.

You should have received a copy of the GNU Library General Public
License along with this library; if not, write to the
Free Software Foundation, Inc., 59 Temple Place - Suite 330,
Boston, MA 02111-1307, USA.

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:



Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

signature of Ty Coon, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

附錄 B Ports 軟體分類列表

目前在 FreeBSD ports 中，共收錄超過六千六百個軟體。以下為目前所有軟體的分類：

分類名稱	數量	說　明
Afterstep	23	Ports to support the AfterStep window manager.
Archivers	57	Utilities for archiving and unarchiving data.
Astro	48	Applications related to astronomy.
Audio	289	Audio utilities - most require a supported sound card.
Benchmarks	25	Utilities for measuring system performance.
Biology	49	Software related to Biology.
Cad	37	Computer Aided Design utilities.
Chinese	92	Ported software for the Chinese market.
Comms	59	Communications utilities.
Converters	66	Format conversion utilities.
Databases	171	Database software.
Deskutils	52	Various Desktop utilities.
Devel	760	Software development utilities and libraries.
Editors	242	Common text editors.
Elisp	157	Emacs lisp ports.
Emulators	88	Utilities for emulating other OS types.
French	8	Ported software for French countries.
Ftp	66	FTP client and server utilities.
Games	441	Various and sundry amusements.
German	19	Ported software for Germanic countries.
Gnome	215	Components of the Gnome Desktop environment.
Graphics	412	Graphics libraries and utilities.
Hebrew	4	Ported software for Hebrew language.
Ipv6	106	IPv6 related software.
Irc	62	Internet Relay Chat utilities.
Japanese	450	Ported software for the Japanese market.
Java	101	Java language support.
Kde	99	Software for the K Desktop Environment.
Korean	71	Ported software for the Korean market.
Lang	245	Computer languages.

分類名稱	數量	說　　明
Linux	173	Linux applications and support utilities.
Mail	285	Electronic mail packages and utilities.
Math	163	Mathematical computation software.
Mbone	14	Applications and utilities for the mbone.
Misc	282	Miscellaneous utilities.
Net	604	Networking utilities.
News	76	USENET News support software.
Offix	6	An office automation suite of sorts.
Palm	30	Software support for the 3Com Palm(tm) series.
Perl5	672	Utilities/modules for the PERL5 language.
Picobsd	1	Ports to support PicoBSD.
Plan9	7	Software from the plan9 Operating System.
Print	292	Utilities for dealing with printing.
Python	170	Software related to the python language.
Ruby	197	Software related to the ruby language.
Russian	28	Ported software for the Russian market.
Science	9	Scientific applications.
Security	307	System security software.
Shells	32	Various shells (tcsh, bash, etc).
Sysutils	240	Various system utilities.
Tcl80	6	TCL v8.0 and packages which depend on it.
Tcl81	1	TCL v8.1 and packages which depend on it.
Tcl82	12	TCL v8.2 and packages which depend on it.
Tcl83	23	TCL v8.3 and packages which depend on it.
Textproc	347	Text processing/search utilities.
Tk42	3	Tk4.2 and packages which depend on it.
Tk80	16	Tk8.0 and packages which depend on it.
Tk82	44	Tk8.2 and packages which depend on it.
Tk83	43	Tk8.3 and packages which depend on it.
Tkstep80	10	The Tk toolkit with NexTSTEP look and packages which depend on it.
Ukrainian	7	Ported software for the Ukrainian market.
Vietnamese	13	Ported software for the Vietnamese market.
Windowmaker	93	Ports to support the WindowMaker window manager.

分類名稱	數量	說　明
Www	498	WEB utilities (browers, HTTP servers, etc).
X11	205	X Window System based utilities.
X11-clocks	35	X Window System based clocks.
X11-fm	29	X Window System based file managers.
X11-fonts	66	X Window System fonts and font utilities.
X11-servers	39	X Window System servers.
X11-toolkits	152	X Window System based development toolkits.
X11-wm	78	X Window System Window Managers.
Zope	10	Software related to the zope platform.

附錄 C 製作 FreeBSD 安裝光碟

如果您想要自行製作 FreeBSD 安裝光碟，您可以自各大 FTP 站台下載 FreeBSD 回來自行燒錄。FreeBSD 各個 RELEASE 版本推出時，都會將光碟的 ISO 檔釋出，如果您要燒錄 RELEASE 版的開機光碟，必須自 ftp 站台下載 ISO 檔。如果您所要燒錄的版本是 STABLE 版，通常都沒有 ISO 檔可以下載，我們必須自行選擇所要燒錄的檔案回來燒錄。在這裡我們就分別針對 RELEASE 及 STABLE 版本的光碟燒錄來加以說明。我們使用的燒錄軟體是 NERO，您可以自 http://www.nero.com 下載試用版本。

C.1 燒錄 RELEASE 版安裝光碟

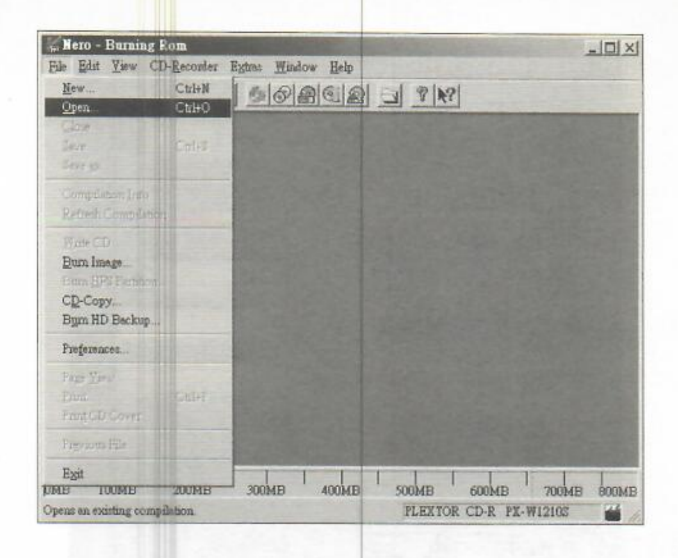
首先，請到離您最近的 FTP 站台下載所需的 ISO 檔，以 FreeBSD 4.5-RELEASE 為例，我們以 FTP 軟體至 ftp://freebsd.csie.nctu.edu.tw/pub/i386/ISO-IMAGES/4.5-RELEASE/ 目錄下，您可以看到以下幾個檔案：

表C-1

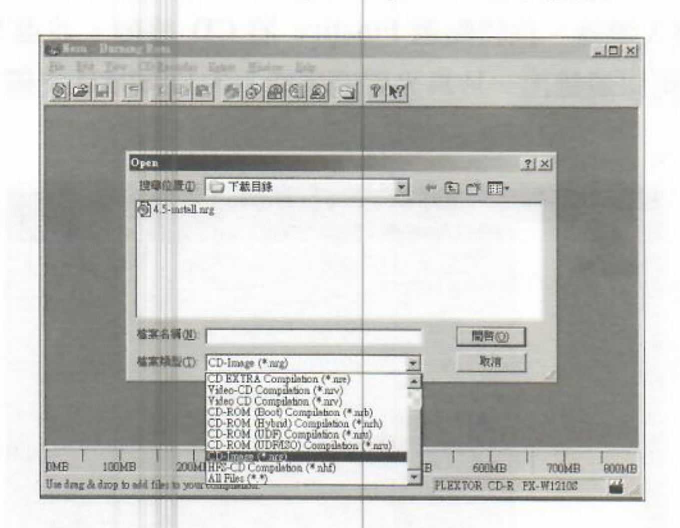
4.5-install.iso	FreeBSD 安裝光碟，包含常用的 packages。
4.5-mini.iso	FreeBSD 安裝光碟，只有安裝必備檔，不含常用的 packages。
4.5-disc2.iso	修復光碟，可以開機並可自光碟執行系統指令。
4.5-disc3.iso	包含更多的 package 軟體。
4.5-disc4.iso	包含更多的 package 軟體。
CHECKSUM.MD5	各個 ISO 檔的 MD5 檢查結果。

我們選擇第一個 INSTALL 光碟下載後，將檔名改成 4.5-install.nrg 後，請依下列步驟進行燒錄：

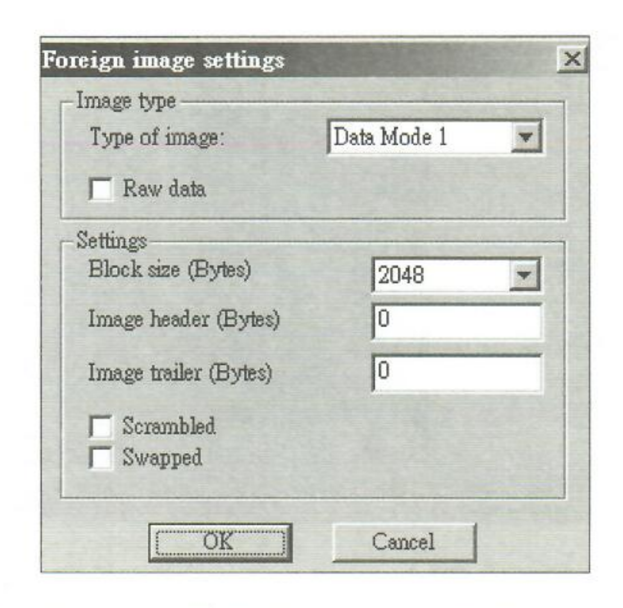
Step1：進入 NERO 後，使用 [File]->[Open] 開啓檔案



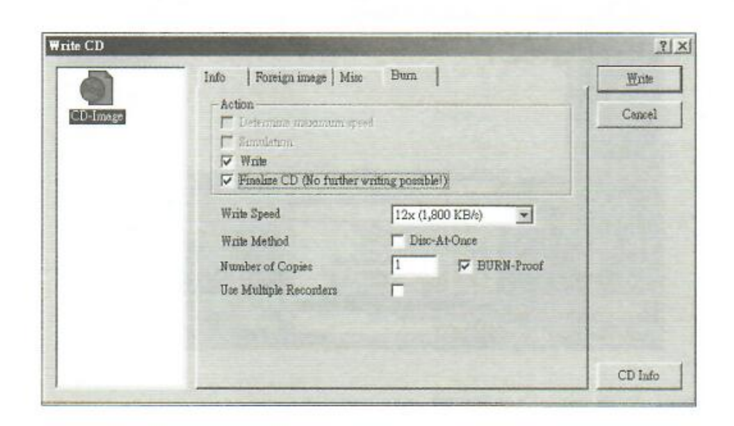
Step2：選擇檔案類型爲 CD-Image(*.nrg)，並選取檔案。



接著將出現下列畫面，直接選 OK 即可：



Step3：進入燒錄。我們點選 Finalize 將 CD 終結，並選 Write 進行燒錄。如此便可以燒錄出一片具光碟開機能力的FreeBSD 光碟。

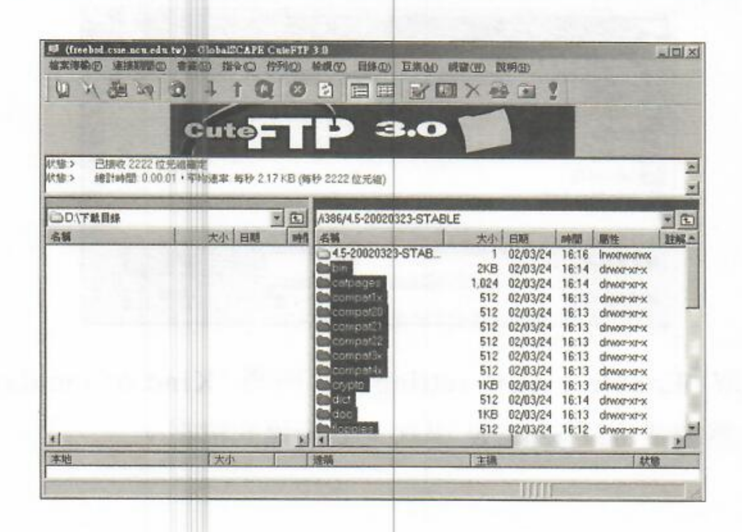


C.2 燒錄 STABLE 版安裝光碟

由於 STABLE 版並沒有 ISO 檔可以下載，因此我們必須自行選擇要下載的檔案。我們以燒錄 FreeBSD 4.5-20020323-STABLE 版為例，請以 FTP 軟體連接到ftp://freebsd.csie.nctu.edu.tw/pub/i386/4.5-20020323-STABLE 目錄下，我們要選取除了下列三個目錄外的所有檔案：

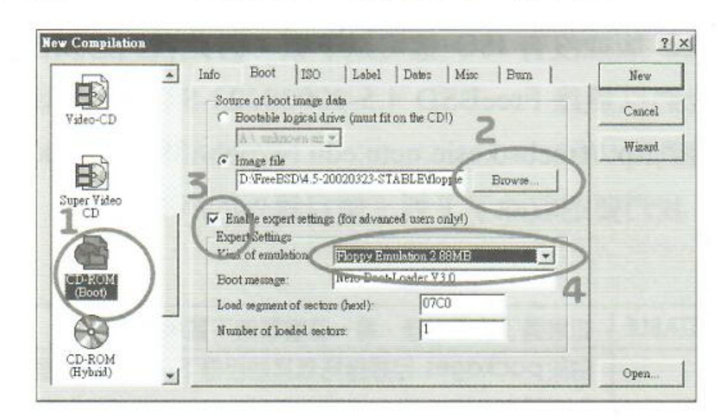
表C-2

4.5-20020323-STABLE	如果選了這個目錄，將一直重覆的下載所有目錄。
packages	由於packages 目錄鏈結到其他目錄下，而且檔案很多，無法完全裝入光碟中。
XF86336	這個目錄也是鏈結到其他目錄下，使用一般 FTP 軟體可能無法順利下載。它的內容通常並未因 RELEASE 或 STABLE 而有所不同，我們可以直接從 RELEASE 光碟中複製該目錄。

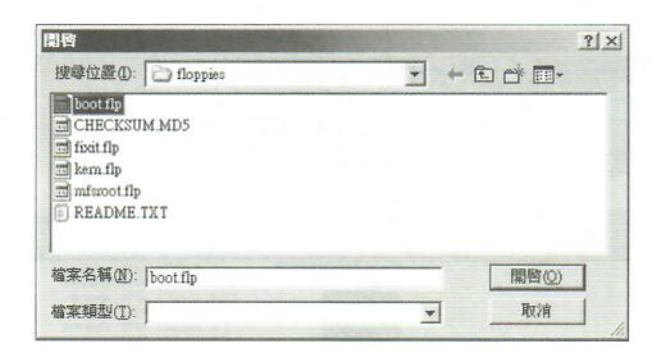


下載完成後，就可以使用下例步驟來燒錄了：

Step1：打開 NERO，選擇燒錄可開機光碟，如下圖：



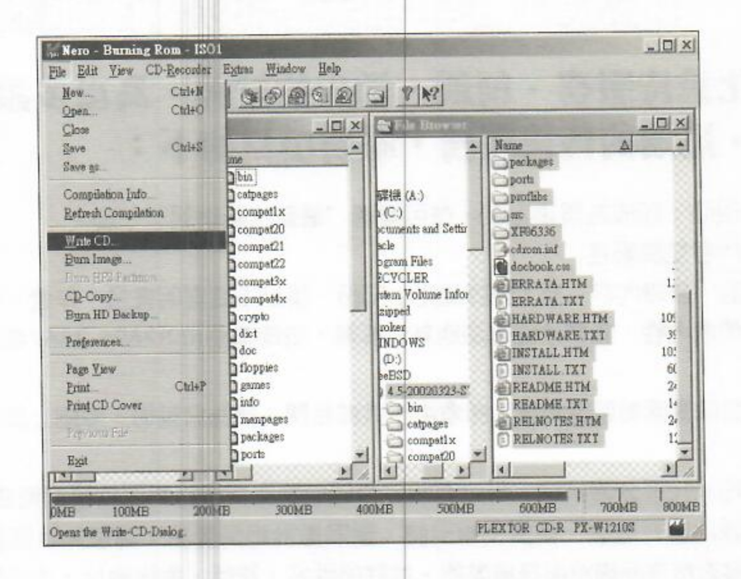
接著選取開機映象檔所在，請先選 Image file，再按 Browse去下載完後 STABLE 版的 floppies 目錄中，選取 boot.flp 這個檔來當開機檔：



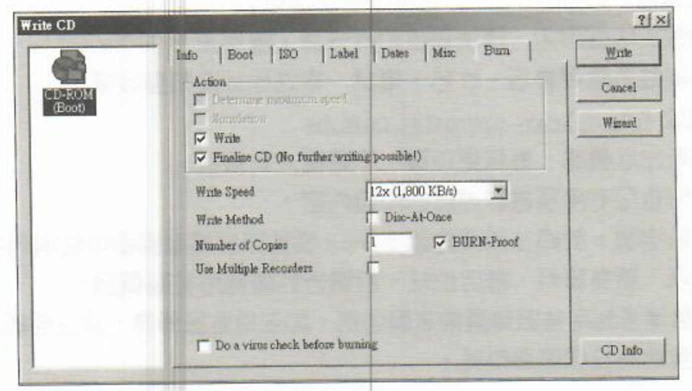
接著選取 "Enable expert settings"，再將 "Kind of emulation" 設爲 2.88MB。然後就可以按 NEW 來加入要燒錄的檔案。



Step2：我們將該來下載的所有檔案加入燒錄清單後，就可以執行寫入了。



Step3：寫入 CD。在上圖中執行 "Write CD" 後，將出現下列畫面，我們直接選取 "Write" 即可進入燒錄。如此便完成 STABLE 版安裝光碟的製作了。





New 網路服務事業部

讓您買書享有完善的售後服務

博碩文化秉持服務、創新、進步的精神，為更多讀者提供更豐富、迅速的售後服務，服務項目如下：

1. 寄回讀者回函：即成為博碩之友，並可獲贈"最新出版情報"一份。
2. 免費書籍內容諮詢服務
   ◎僅限於對「書籍內容」及「光碟使用」部份，請以正楷寫明書名、作者、頁碼，並詳述您的問題內容，我們才能清楚地為您解答，請傳真至(02)2696-2867或利用客服信箱。
   ◎在收到您的書面問題資料後，將會由作者或相關人員為您說明，原則上於一週內為您答覆。
   ◎因本公司未代理軟體銷售，所以有關軟體問題請直接連絡相關軟體廠商或代理商。
3. 換書：書籍內容有瑕疵、毀損及短少時，僅限更換相同產品。請您先以客服電話登記後，將該書直接寄回網路服務事業部，並註明姓名、電話、寄送地址，本公司將於收到書籍後一週內將新書補寄予您。
4. 附件遺失補發：請附購買發票或收據証明影本傳真至(02)2696-2867
   ◎光碟類【劃撥費用 100 元(內含光碟 + 掛號費)】
   ◎磁片類【填寫 E-mail 帳號，方便為您傳送檔案】
   ◎部份書籍限於版權限定，無法提供附件分售，請見諒。
   ◎傳真資料請註明購買書名、姓名、電話、寄送地址及劃撥收據。
5. 博碩文化網址 http://www.drmaster.com.tw
   ◎提供您最新出版書訊、書籍更正區、下載區、留言版。
   ◎留言版：您也可利用留言詳述您的問題內容。
   ◎免費電子報申請：於網站首頁登錄 E-mail 帳號或於讀者回函中註明均可。
6. 各式查詢服務：新書資料、書店位址、訂購方式請利用客服電話。
   以上服務項目請多利用客服傳真或客服信箱，如使用客服傳真，請註明網路服務事業部李琬茹收，我們將立即與您聯絡。

服務時間：週一到週五 08:30am~5:30pm
客服電話：(02)2696-2869 分機 213 李琬茹
客服傳真：(02)2696-2867
客服信箱：Web@drmaster.com.tw
公司地址：221 台北縣汐止市新台五路一段 112 號 10 樓 A 棟

